



US006446076B1

(12) **United States Patent**
Burkey et al.

(10) Patent No.: **US 6,446,076 B1**
(45) Date of Patent: ***Sep. 3, 2002**

(54) **VOICE INTERACTIVE WEB-BASED AGENT SYSTEM RESPONSIVE TO A USER LOCATION FOR PRIORITIZING AND FORMATTING INFORMATION**

(75) Inventors: **Chad Burkey, Atherton; Sergei Lopatin, Mountain View; Lucian Hughes, Montara, all of CA (US)**

(73) Assignee: **Accenture LLP, Palo Alto, CA (US)**

(*) Notice: This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/196,482**

(22) Filed: **Nov. 19, 1998**

Related U.S. Application Data

(60) Provisional application No. 60/115,515, filed on Nov. 12, 1998.

(51) Int. Cl.⁷ **G06F 17/30**

(52) U.S. Cl. **707/102; 709/229; 704/270**

(58) Field of Search **704/275, 270; 709/218, 229, 219; 713/201; 345/335; 707/501, 2, 513, 102**

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,278,943 A * 1/1994 Gasper et al. 704/200
5,481,255 A * 1/1996 Albert et al. 340/825.55
5,493,692 A * 2/1996 Theimer et al. 455/26.1
5,694,459 A * 12/1997 Backaus et al. 379/127

(List continued on next page.)

FOREIGN PATENT DOCUMENTS

| | | | | |
|----|------------|---------|-------|------------|
| EP | 0828222 A1 | 3/1998 | | G06F/17/30 |
| WO | 97/41673 | 11/1997 | | 705/10 |
| WO | 98/03907 | 1/1998 | | 706/45 |
| WO | 98/21872 | 5/1998 | | 379/88.22 |
| WO | 98/35469 | 8/1998 | | 705/14 |
| WO | 98/40832 | 9/1998 | | 705/10 |

OTHER PUBLICATIONS

Marcus, Aaron and Andries van Dam, "User-Interface Developments for the Nineties," Computer, vol. 24, No. 9, Sep. 1991, pp. 49-57.*

(List continued on next page.)

Primary Examiner—Marsha D. Banks-Harold

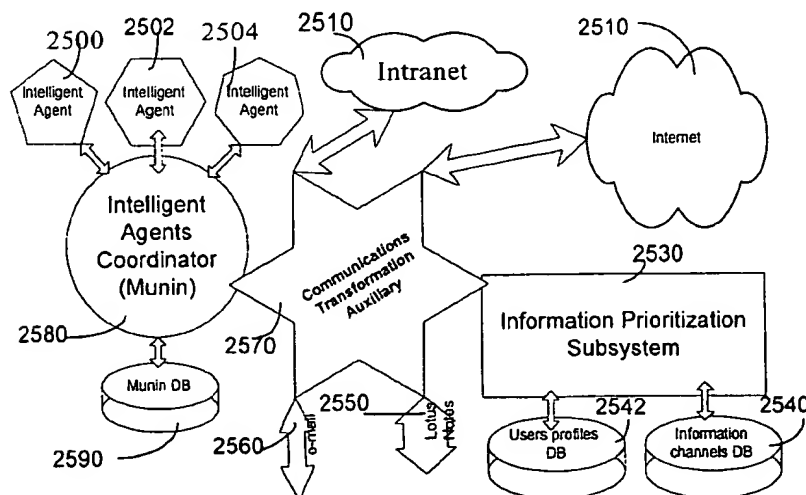
Assistant Examiner—Donald L. Storm

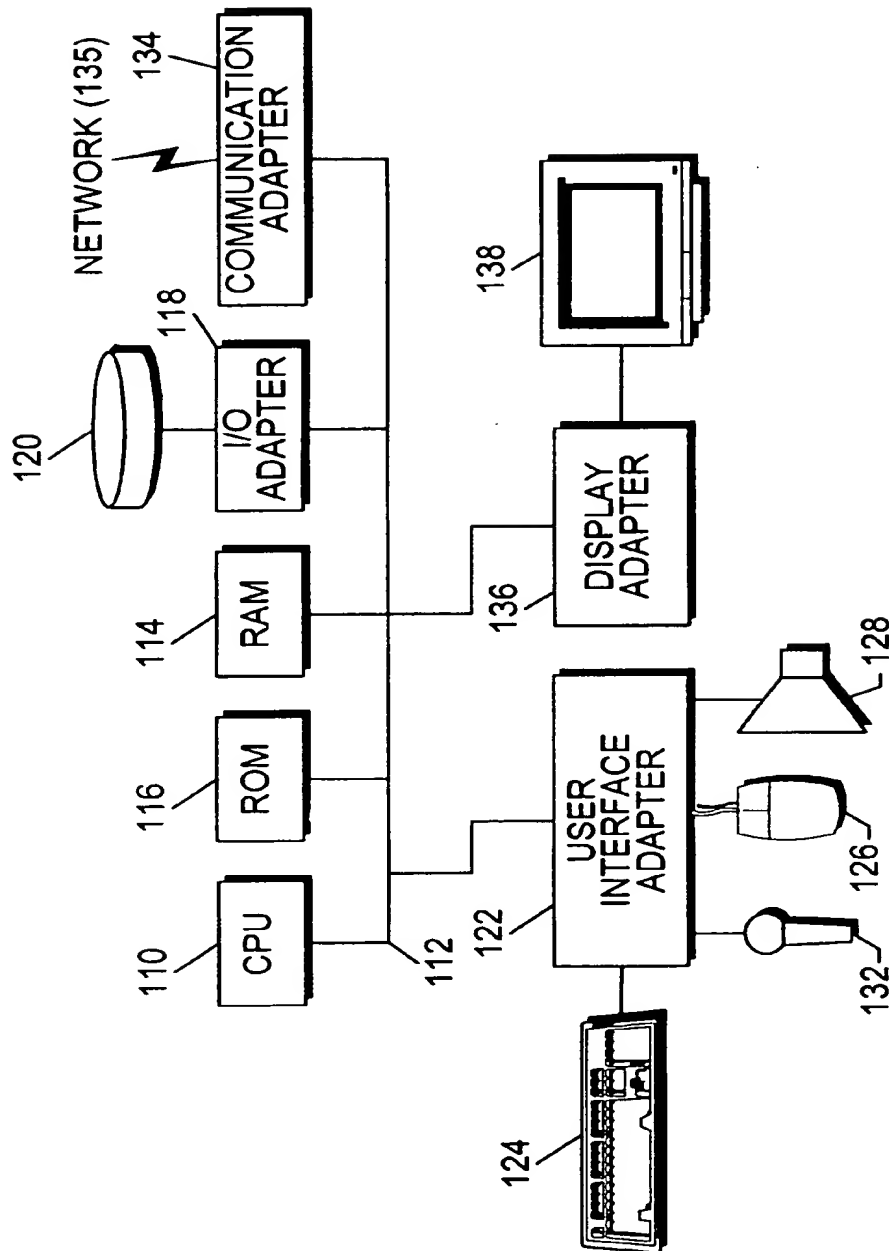
(74) Attorney, Agent, or Firm—Brinks Hofer Gilson & Lione

ABSTRACT

A system, method and article of manufacture for facilitating a user in, or creating for a user, a web-based active knowledge management system to facilitate an intelligent agent coordinator. The architecture facilitates delivery of information whenever and where ever a user requires the information in an appropriate format based on characteristics of the user at that instant. Personalization of information is also afforded by taking into account the history of user interactions with various applications and current real time situations, such a time and place, in order to create a plurality of unique user profile. Each unique profile can then be associated with a unique address, as well as being given restriction exclusive to that profile. Associations, such as business rules or databases, can also be related to each such profile. A fast and scalable information prioritization subsystem is also utilized to incorporate intelligent agents coordinator opinion, user preferences, and history of user interactions. In addition, speech recognition and speech synthesis are combined with intelligent agent animated representation and tactile input.

22 Claims, 27 Drawing Sheets



**FIG. 1**

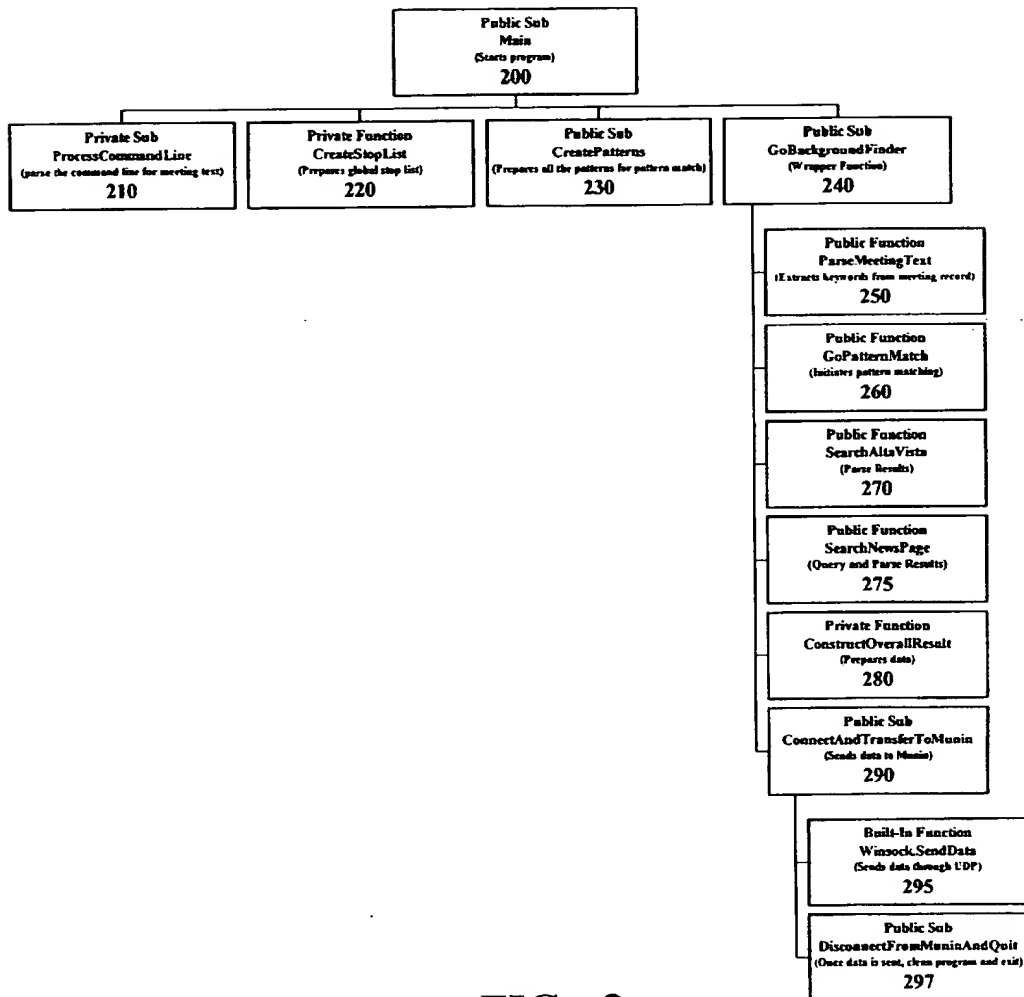


FIG. 2

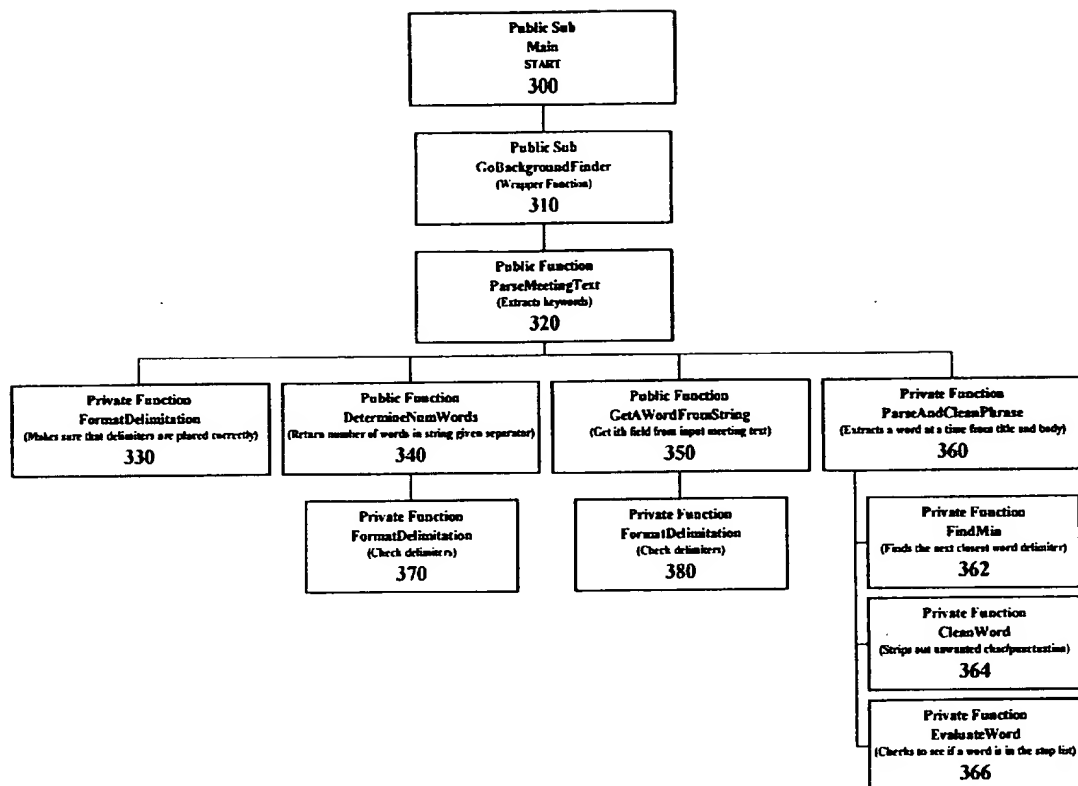


FIG. 3

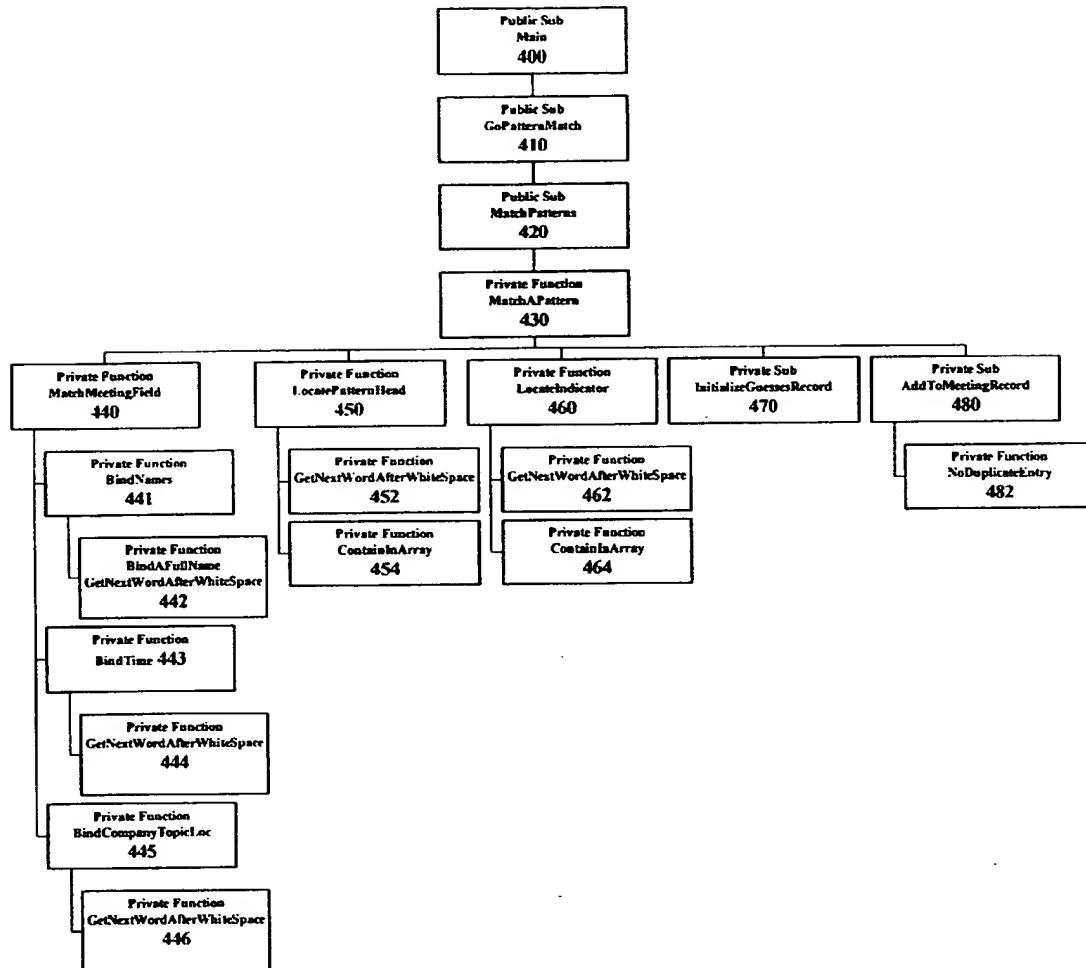


FIG. 4

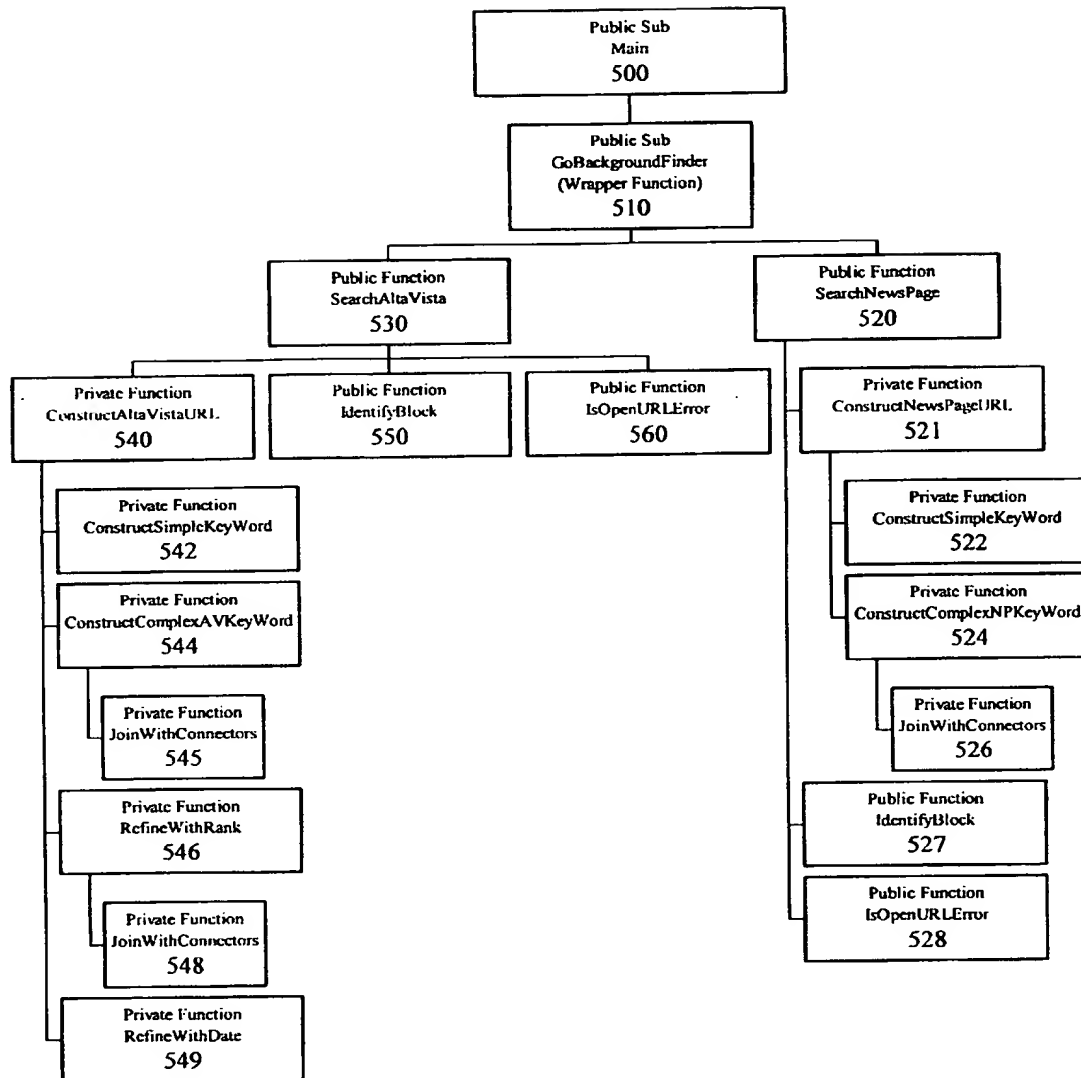
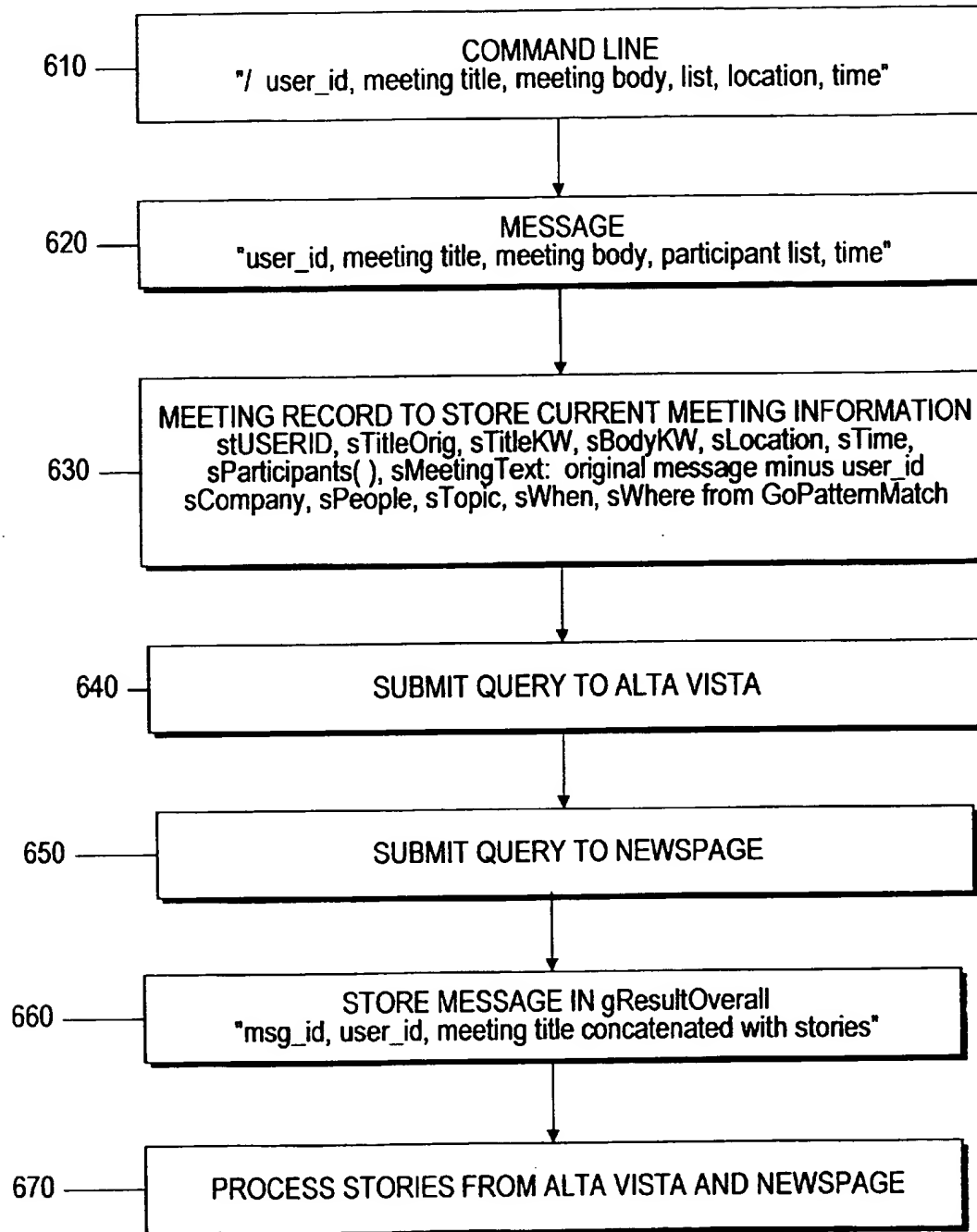
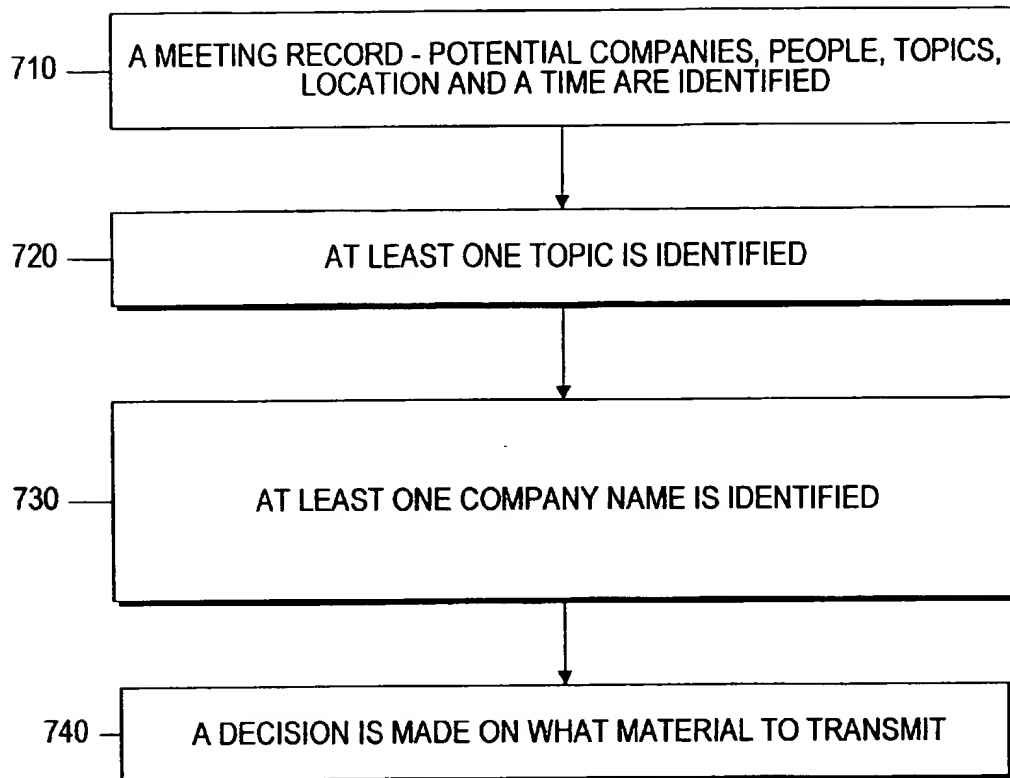
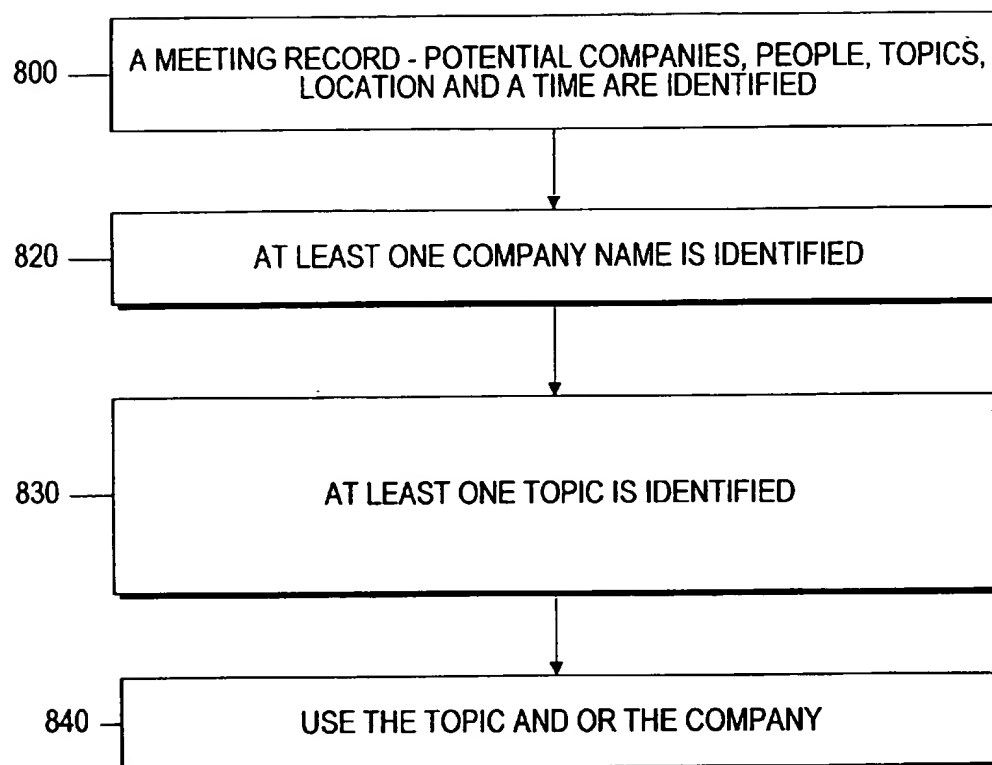


FIG. 5

**FIG. 6**

**FIG. 7**

***FIG. 8***

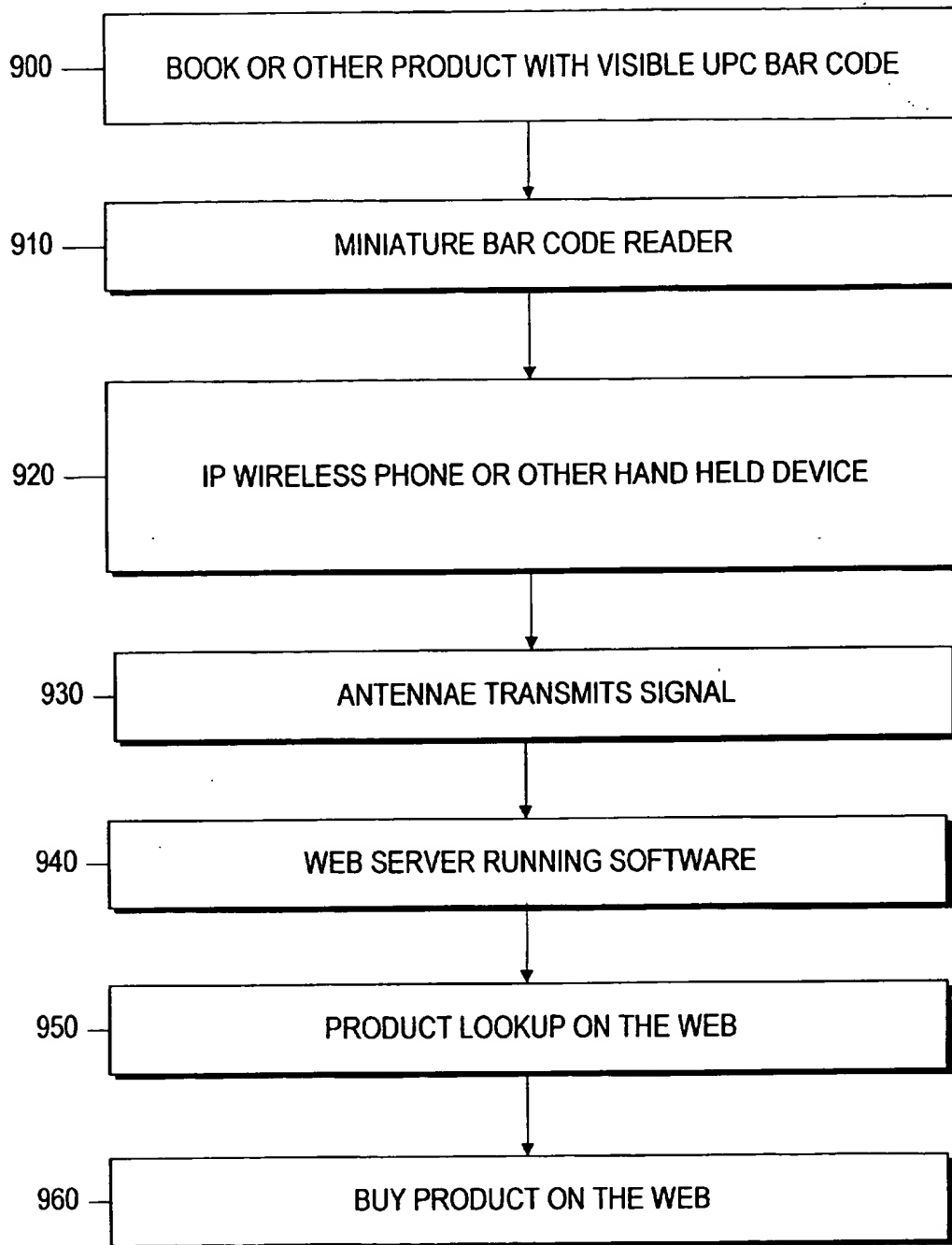
**FIG. 9**

FIG. 10A

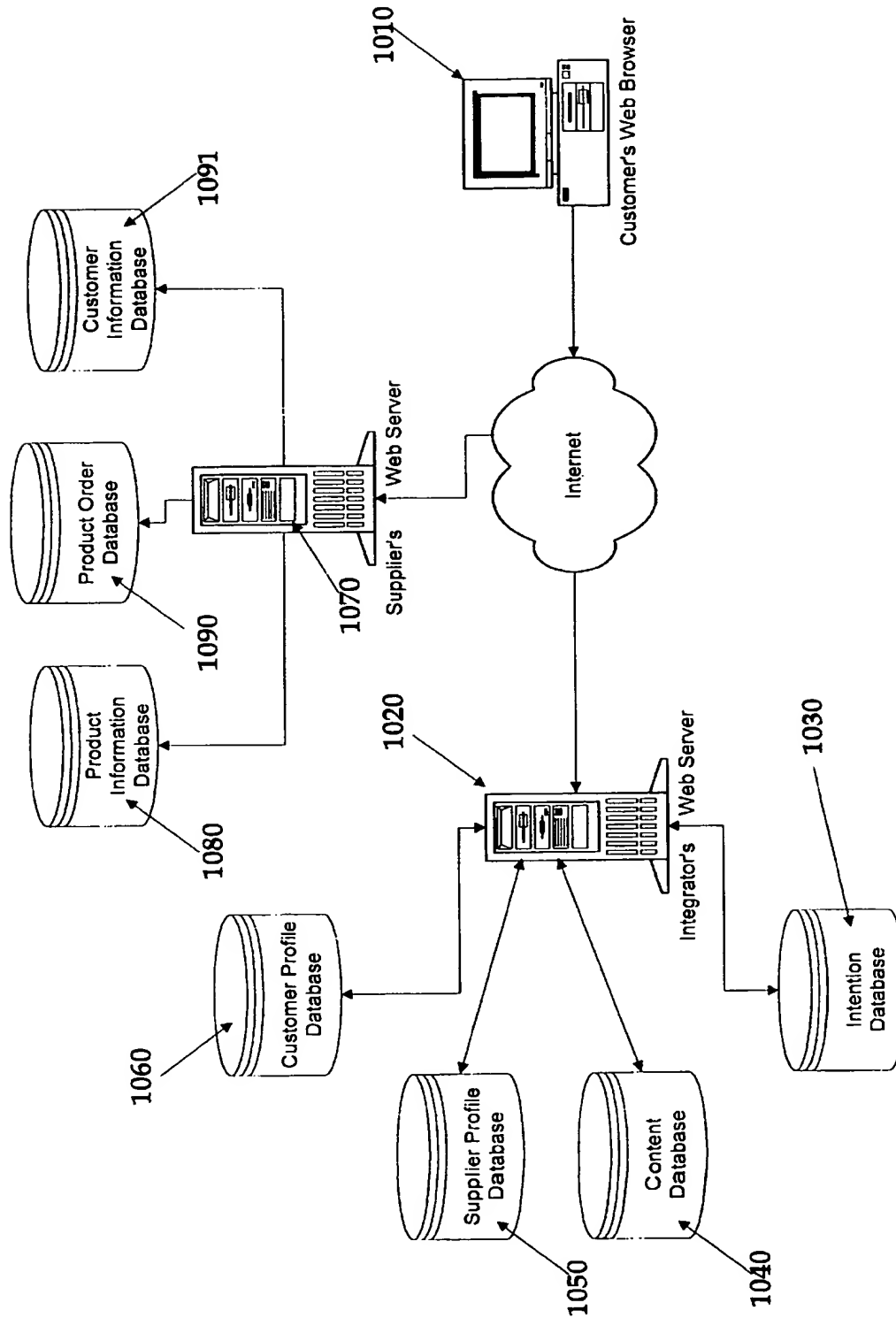


Fig. 10B

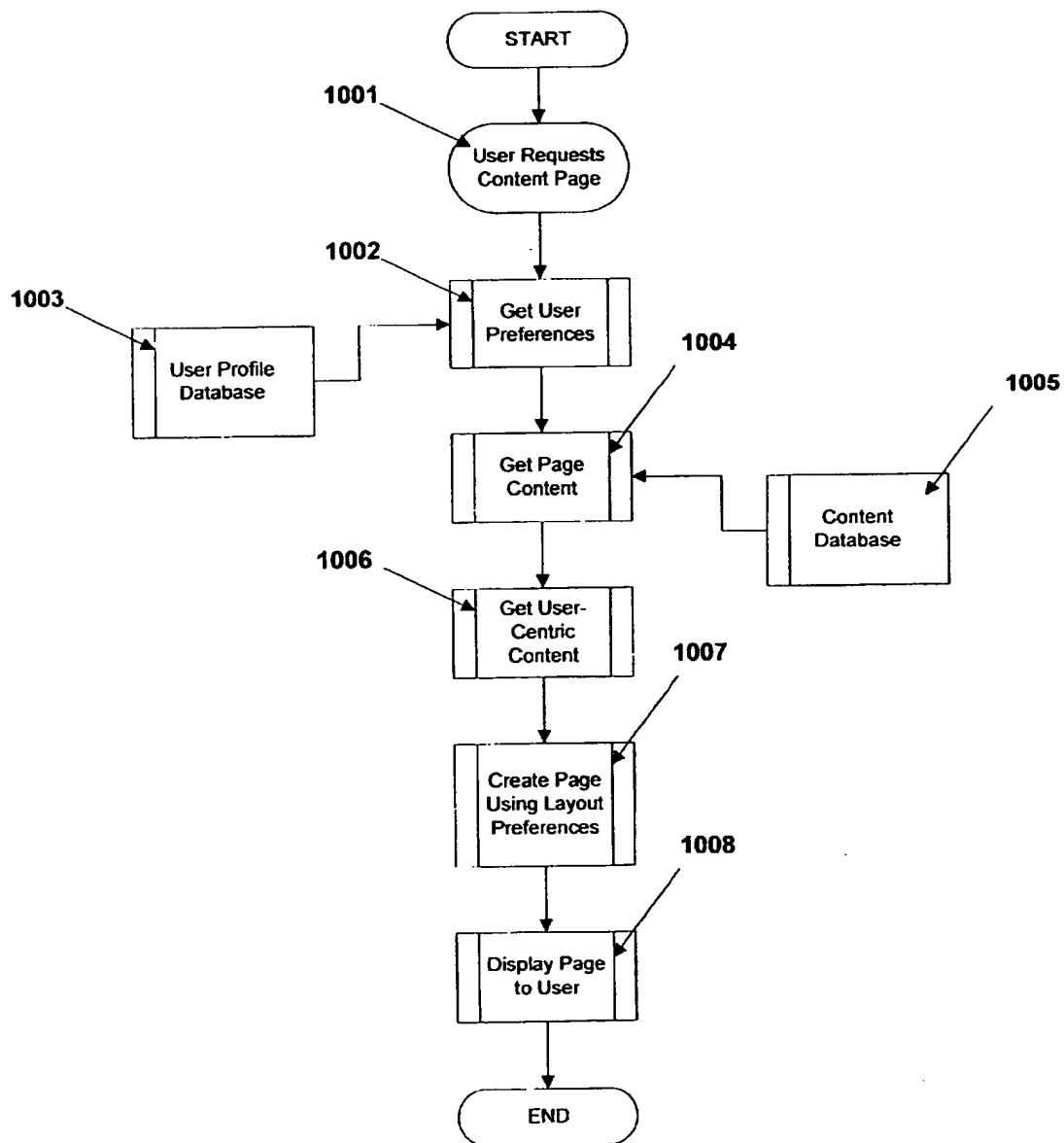


Fig. 11

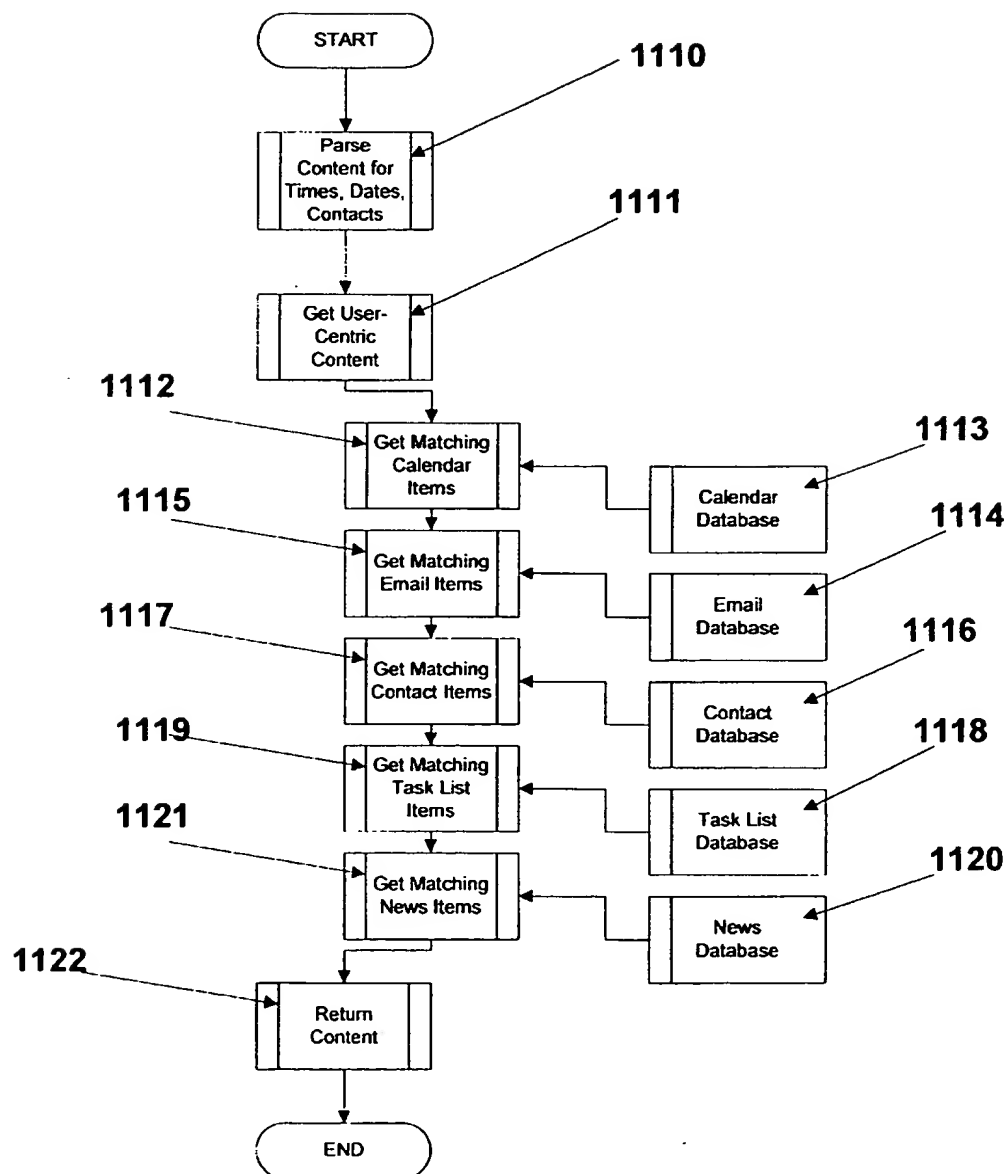


Fig. 12

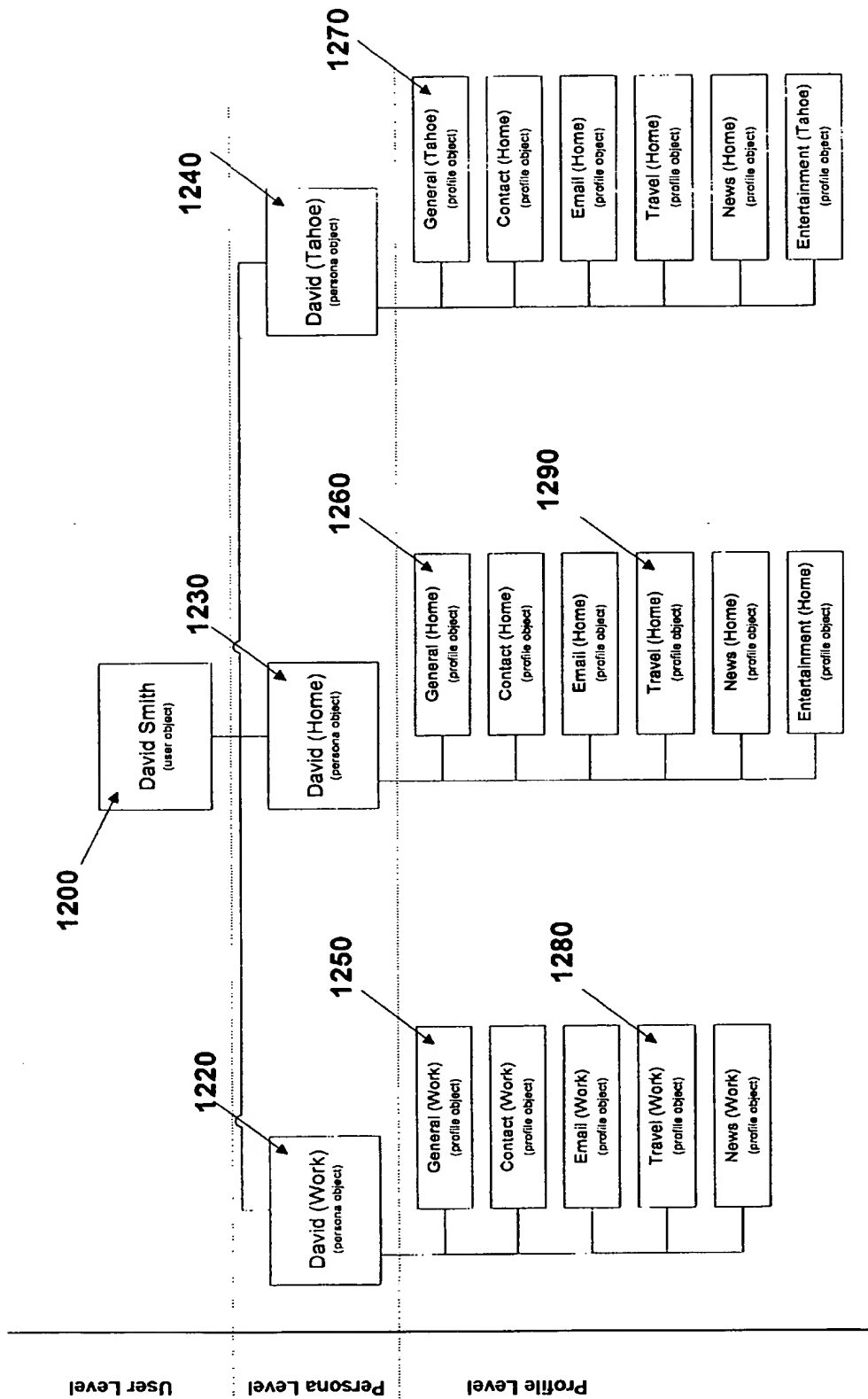


Fig. 13

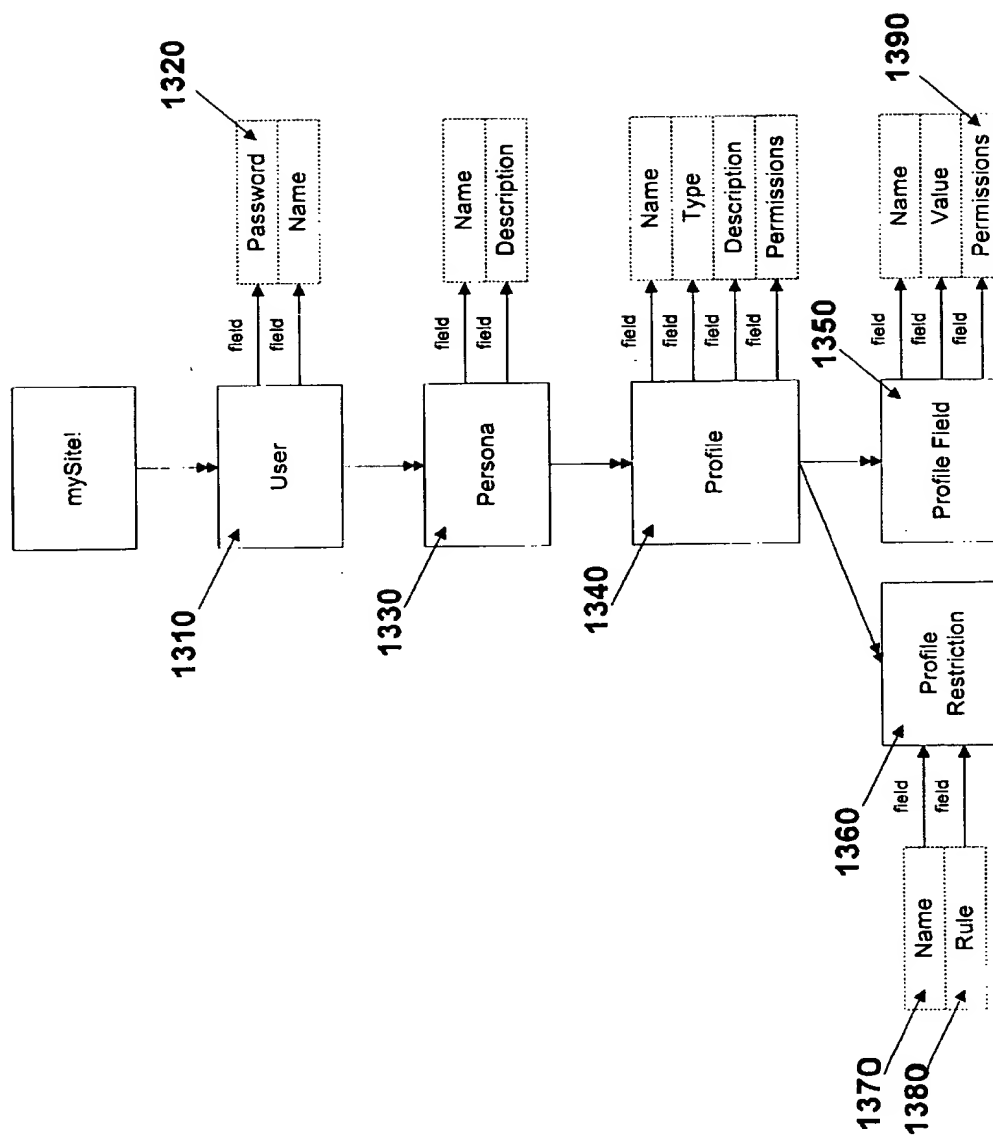


Fig. 14

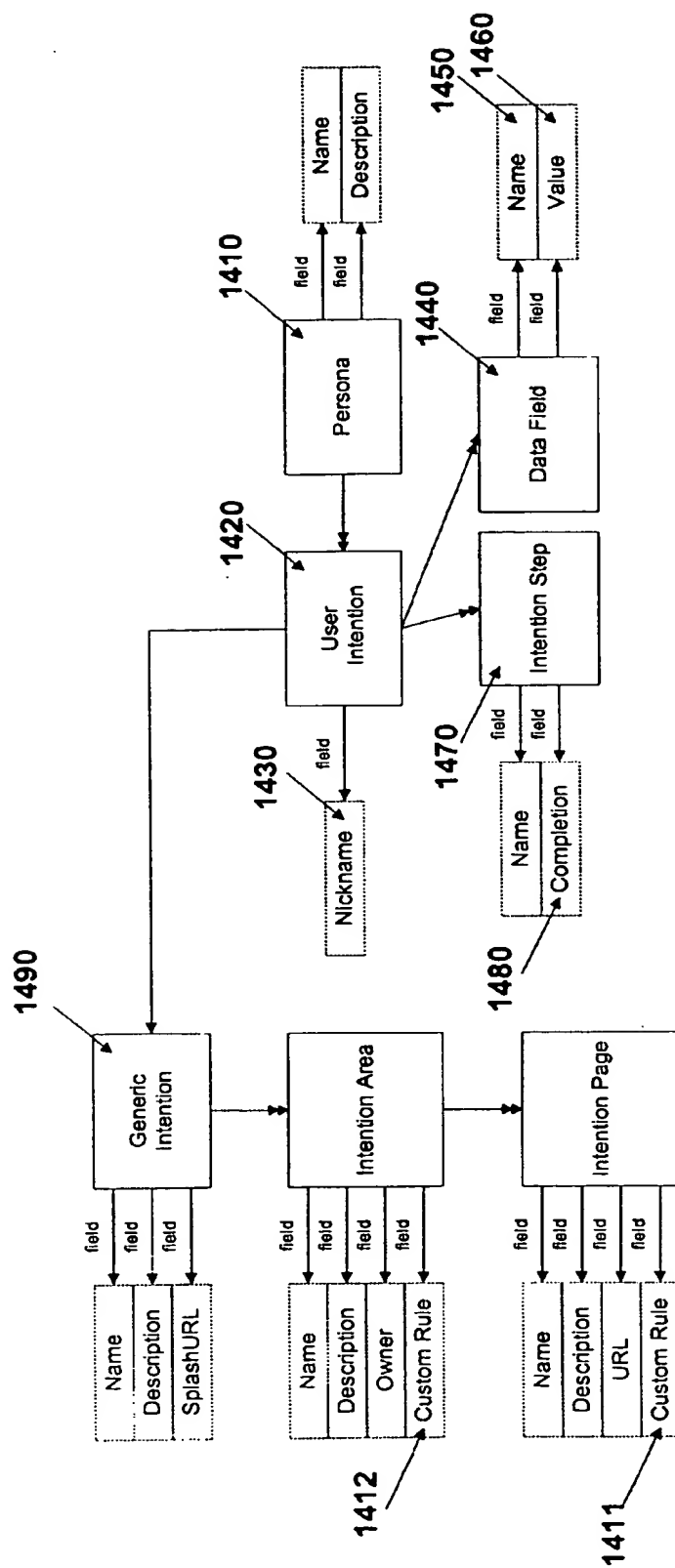


Fig. 15

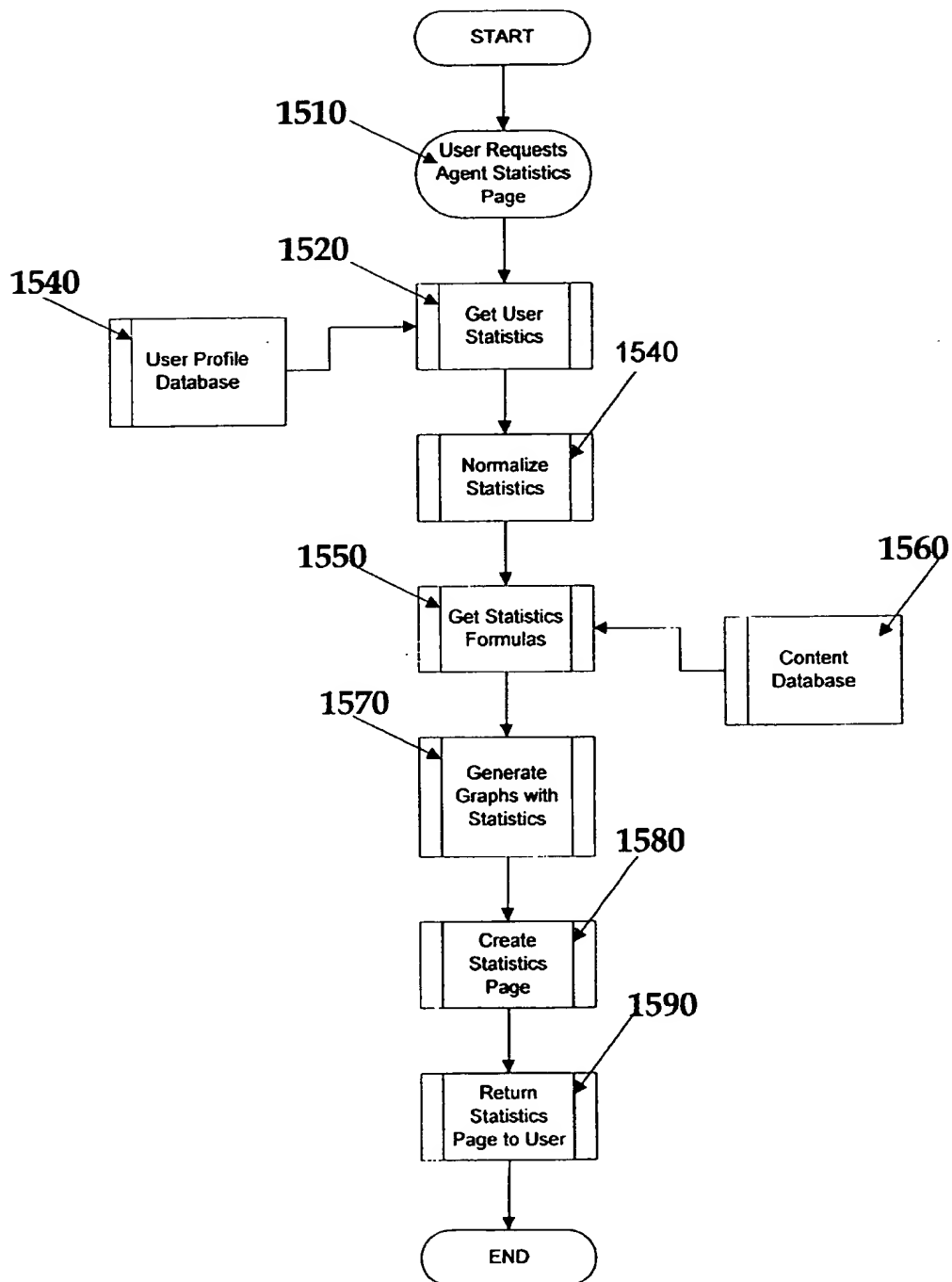


Fig. 16

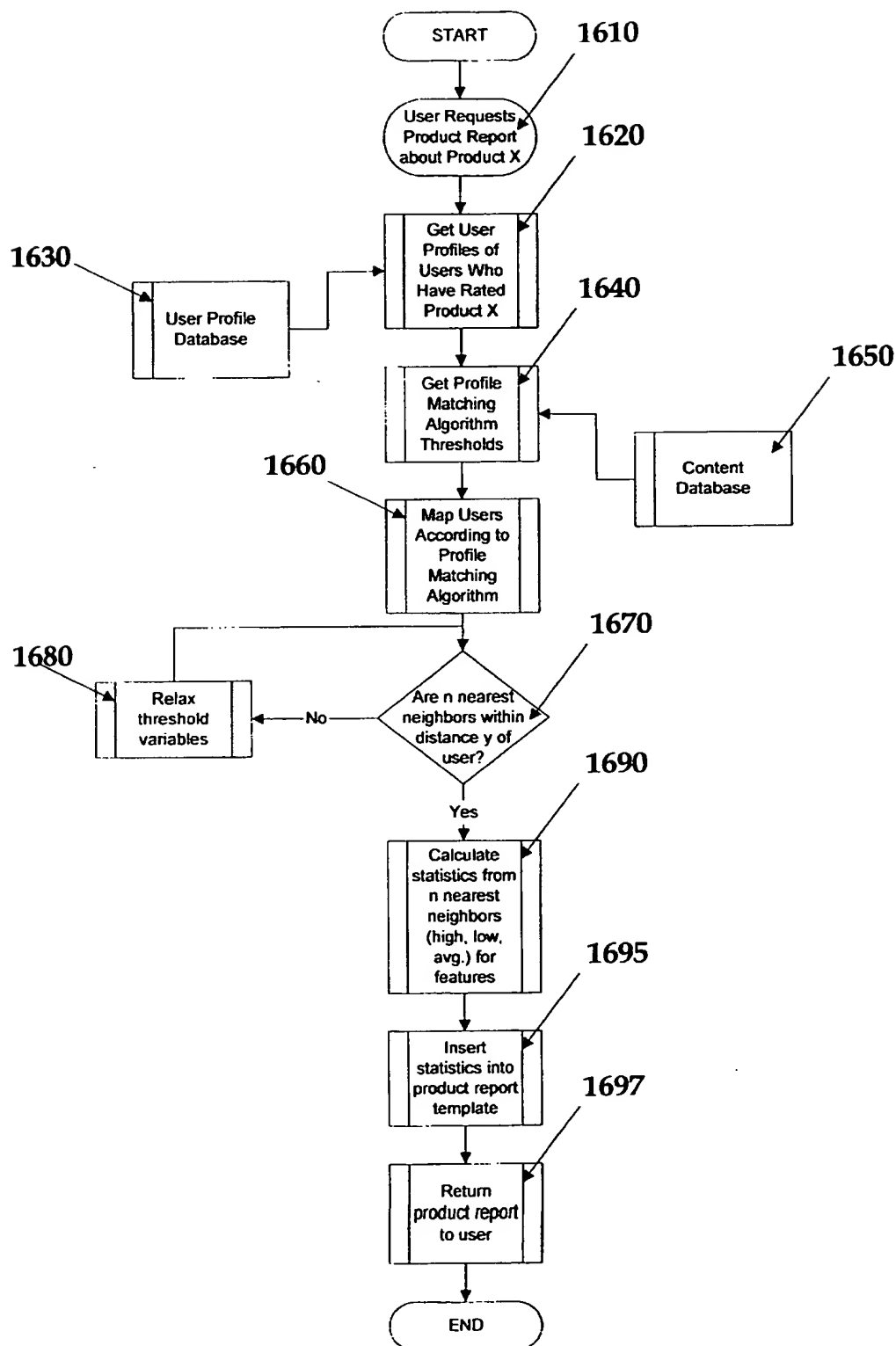


Fig. 17

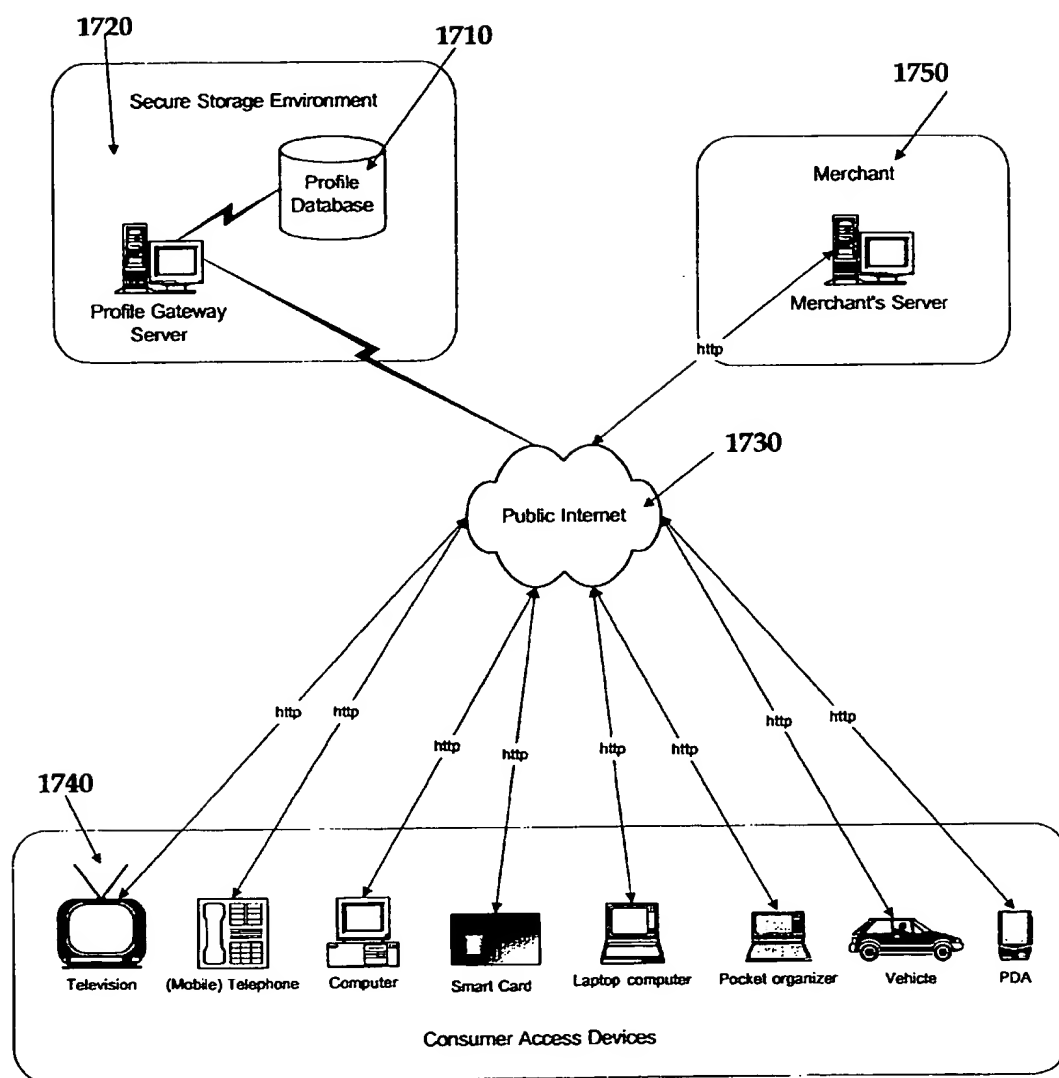


Fig. 18

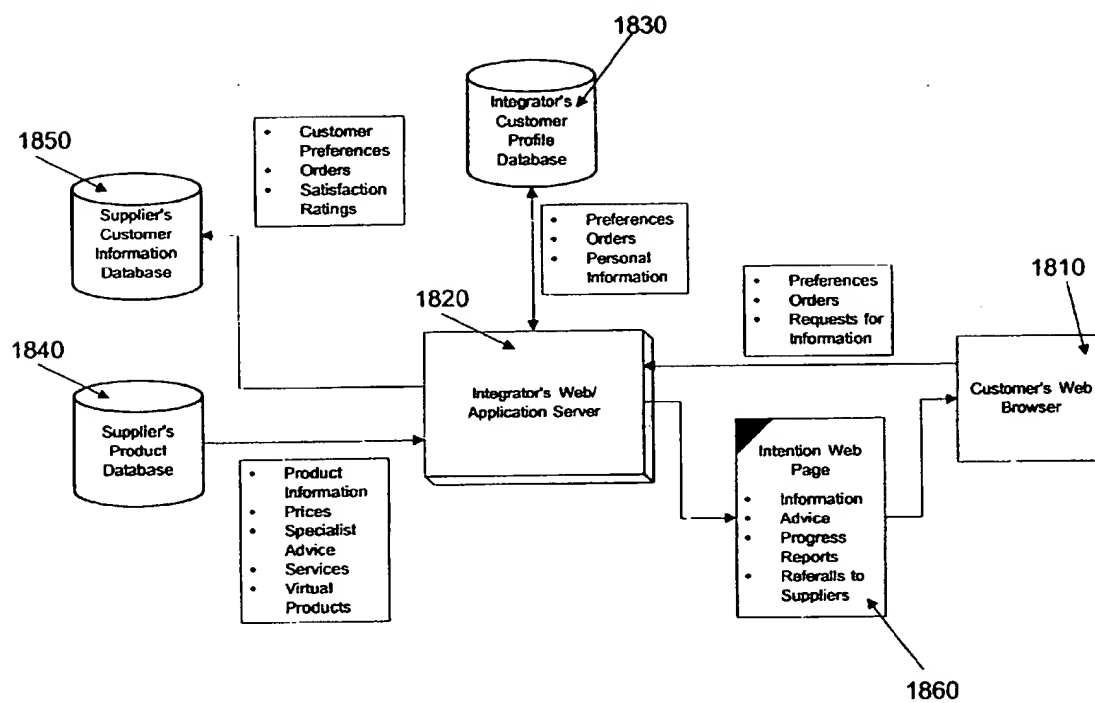
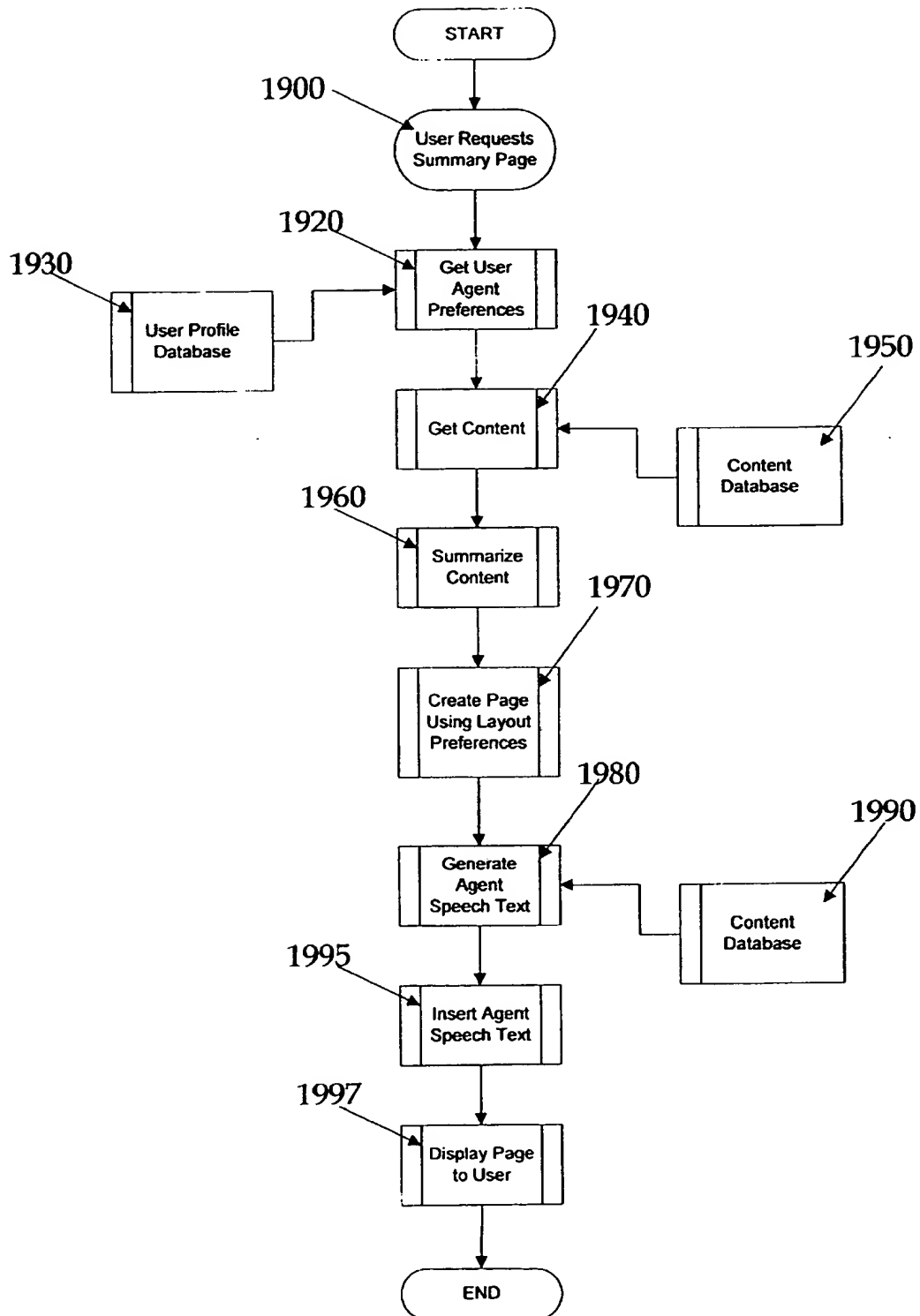


FIG. 19



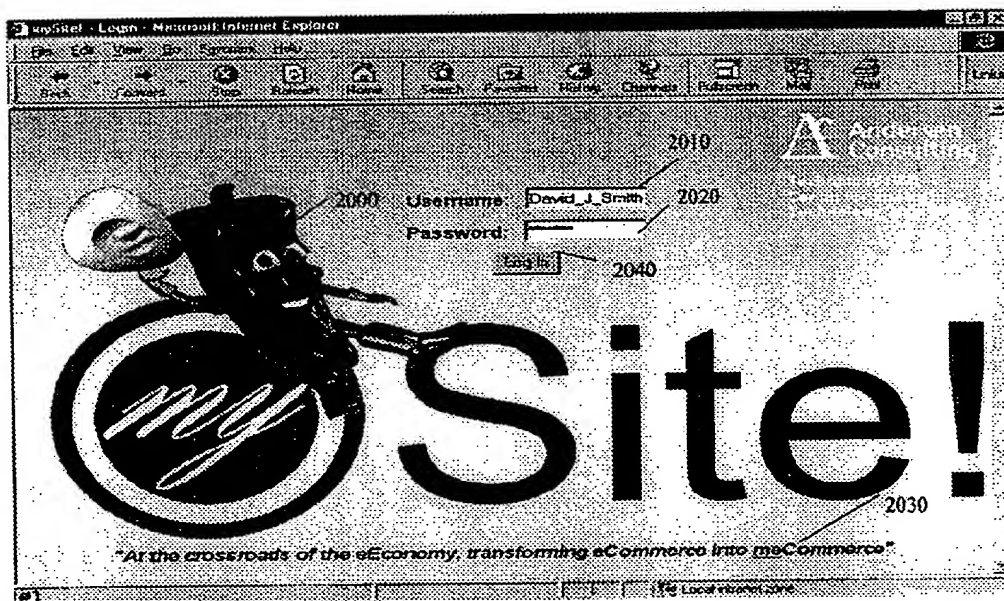


FIG. 20

Best Available Copy

MySite! - David Smith - Microsoft Internet Explorer

File Edit View Go Window Help

Back Forward Stop Refresh Home Search Favorites History Channels Favorites

Welcome back, David!

Managing Daily Logistics

David J. Smith

2142 WORKPLACE 2140 FINANCES 2130 HOUSEHOLD 2120 TRAVEL 2110

Profile Name: My Home Profile Add

2180 2150

First Name David Never Ask Always

Middle Init. J. Never Ask Always

Last Name Smith Never Ask Always

Gender Male Female Never Ask Always

Address 1 1661 Page Mill Road Never Ask Always

Address 2 Apt 300 Never Ask Always

City Palo Alto Never Ask Always

State CA Never Ask Always

Country United States Never Ask Always

Zip Code 94304 Never Ask Always

Homeowner Own Rent Never Ask Always

Personal Information

Financial

Interests

Family

Travel

News

Preferences

Public Page

2146

Local Internet Info

FIG. 21

Available Copy

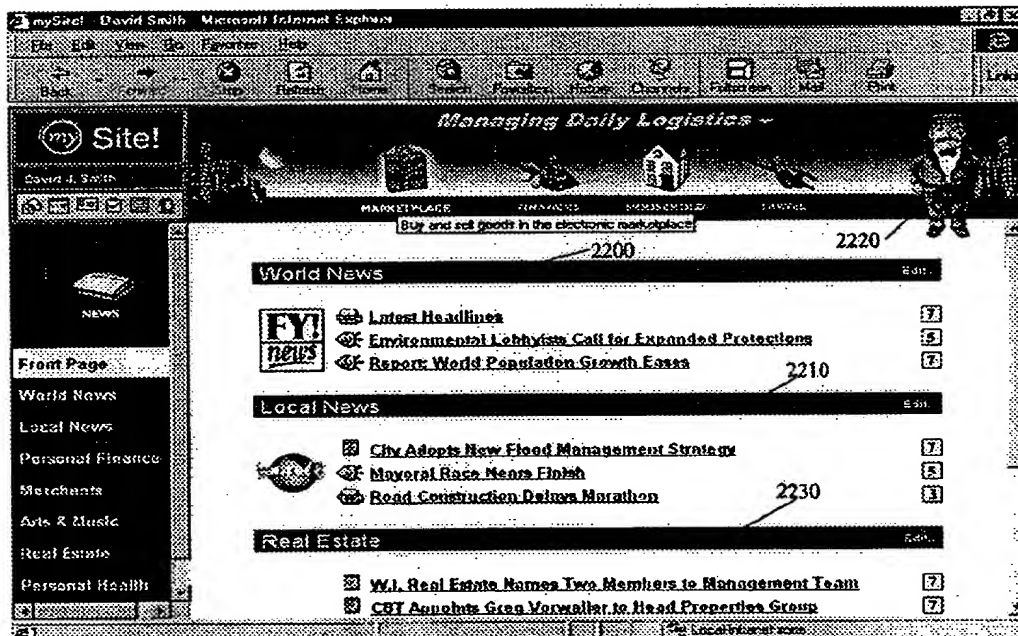
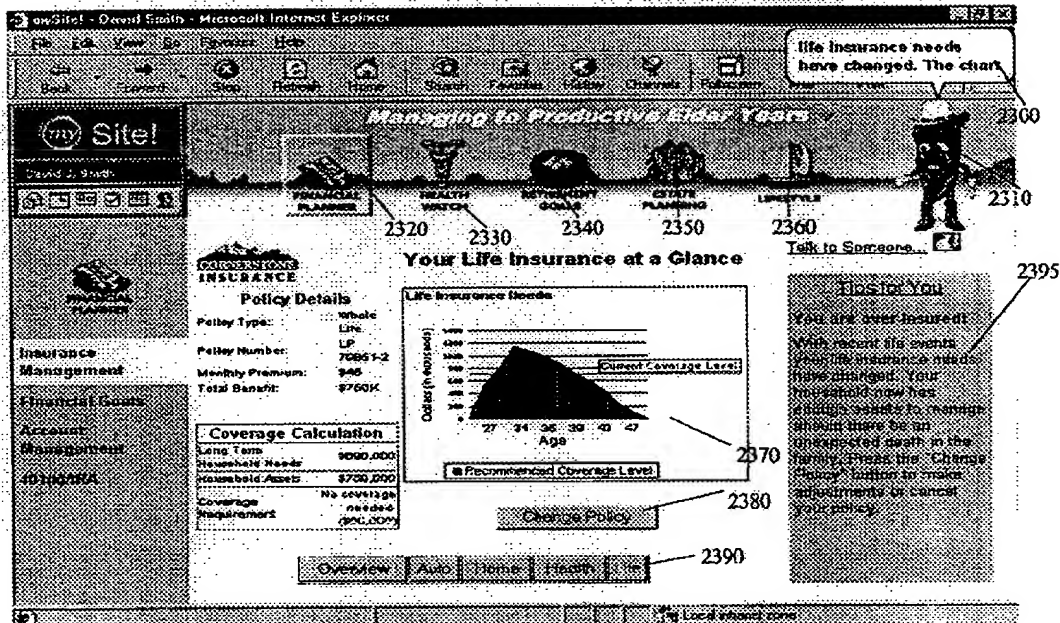
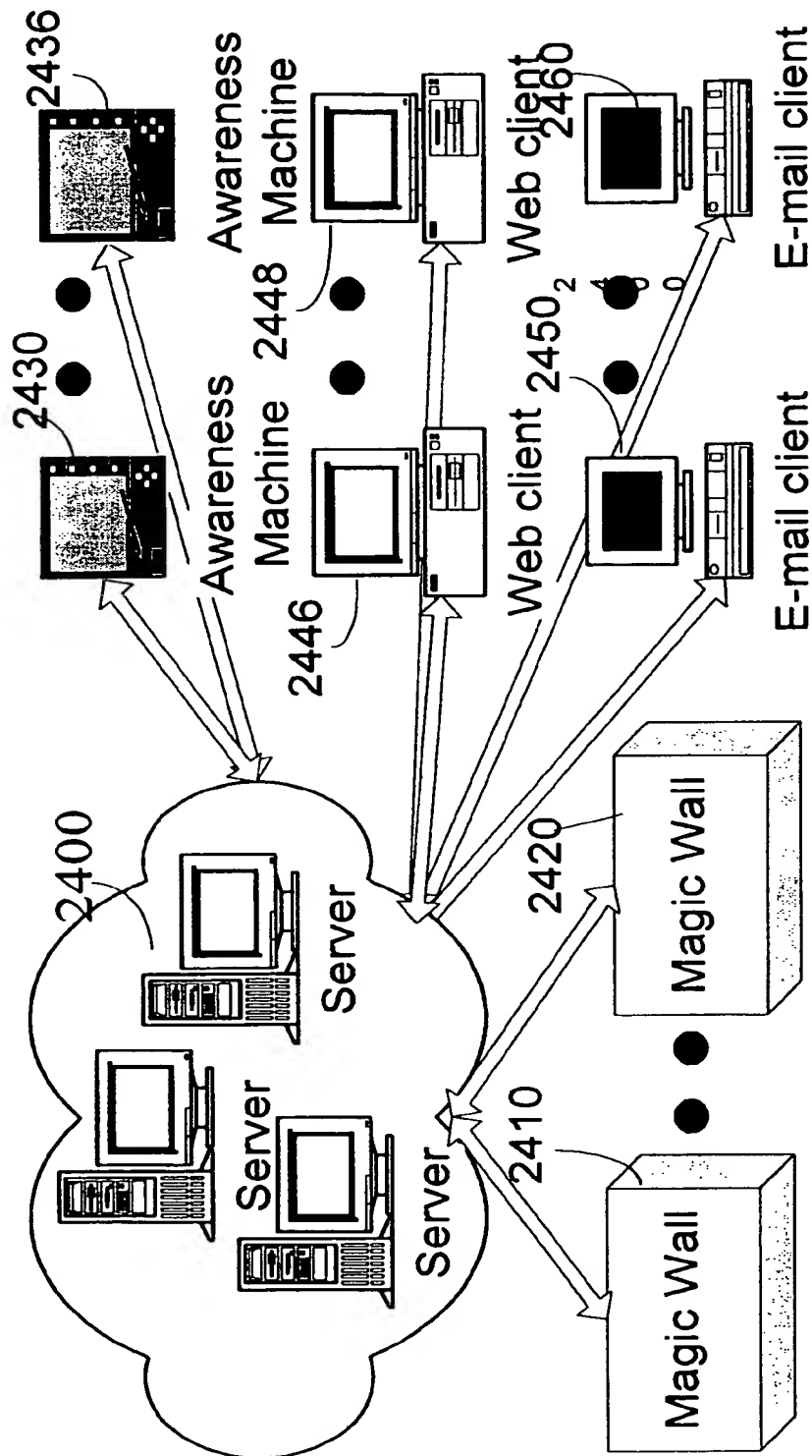


FIG. 22

Best Available Copy



Best Available Copy

**Figure 24**

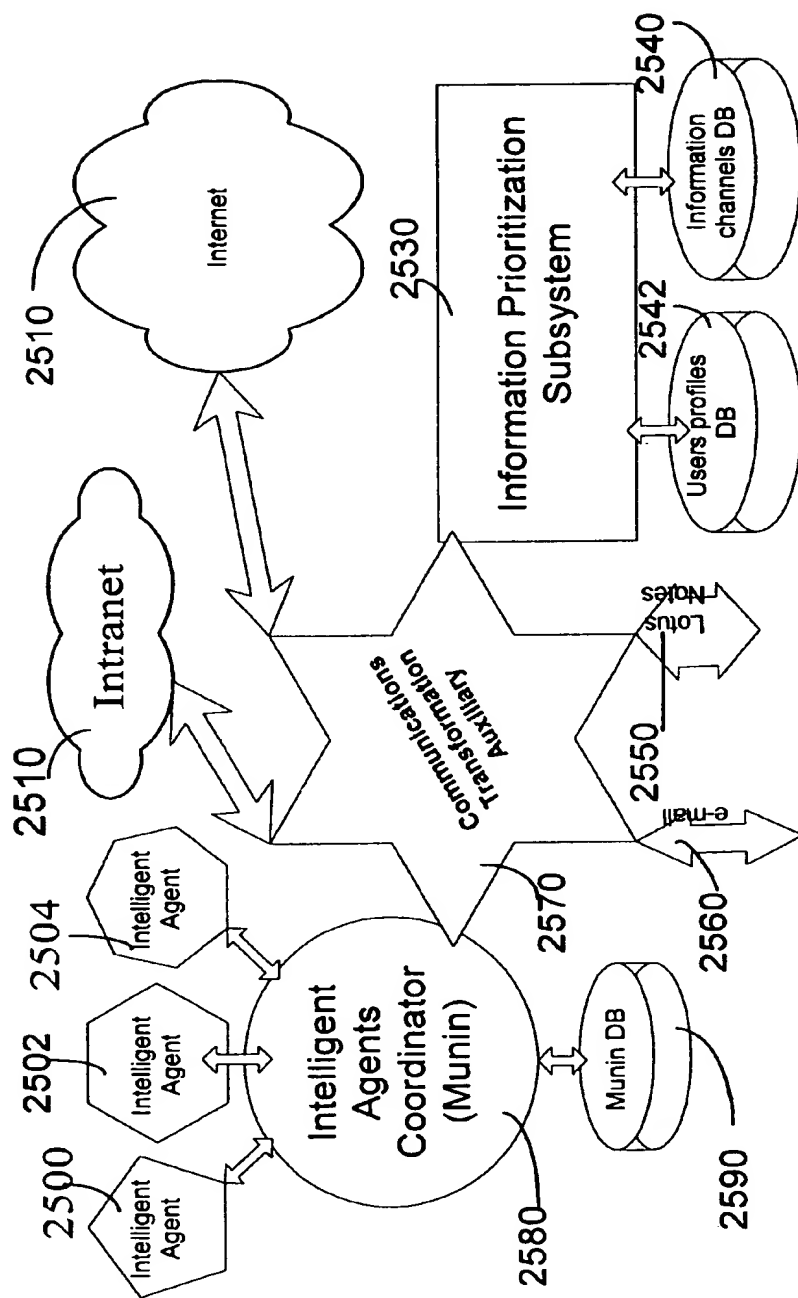


Fig. 25

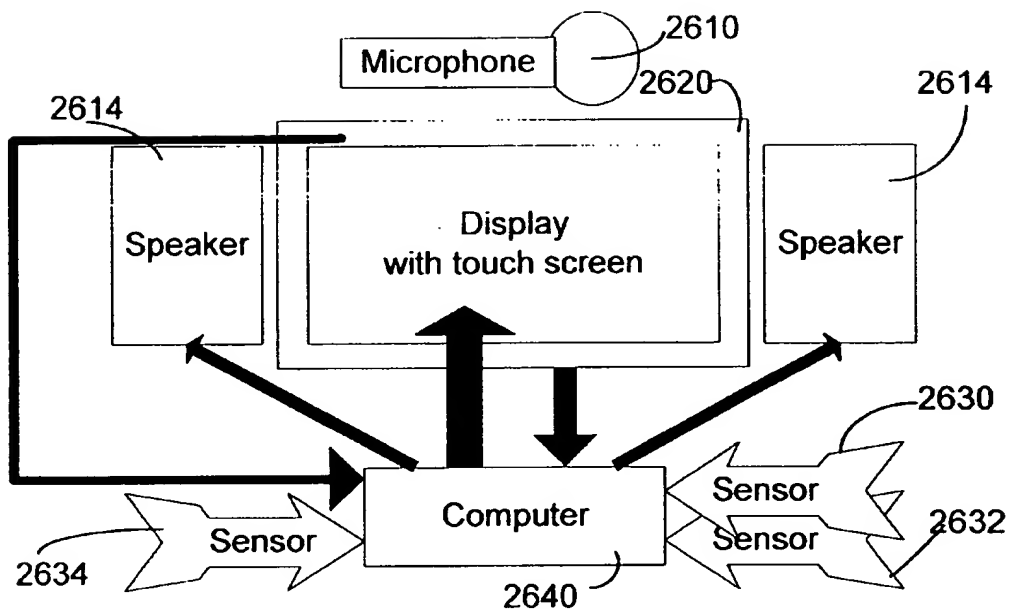


Figure 26

740, a decision is made on what material to transmit to the file for ultimate consumption by the user.

FIG. 8 is a variation on the query theme presented in FIG. 7. A meeting record is parsed in function block 800, a company is identified in function block 820, a topic is identified in function block 830 and finally in function block 840 the topic and or the company is utilized in formulating the query.

Alternative embodiments for adding various specific features for specific user requirements are discussed below.

Enhance Target Rate for Pattern Matching

To increase BF's performance, more patterns/pattern groups are added to the procedure "CreatePatterns." The existing code for declaring patterns can be used as a template for future patterns. Because everything is stored as dynamic arrays, it is convenient to reuse code by cutting and pasting. The functions BindName, BindTime, BindCompanyLoc-Topic which are responsible for associating a value with a placeholder can be enhanced. The enhancement is realized by increasing the set of criteria for binding a certain meeting field in order to increase the number of binding values. For example, BindTime currently accepts and binds all values in the form of ##:## or #:##. To increase the times we can bind, we may want BindTime to also accept the numbers 1 to 12 followed by the more aesthetic time terminology "o'clock." Vocabulary based recognition algorithms and assigning an accuracy rate to each guess BF makes allowing only guesses which meet a certain threshold to be valid.

Depending on what location the system identifies through pattern matching or alternatively depending on what location the user indicates as the meeting place, a system in accordance with a preferred embodiment suggests a plurality of fine restaurants whenever it detects the words lunch/dinner/breakfast. We can also use a site like company finder to confirm what we got is indeed a company name or if there is no company name that pattern matching can identify, we can use a company finder web site as a "dictionary" for us to determine whether certain capitalized words represent a company name. We can even display stock prices and breaking news for a company that we have identified.

Wireless Bargain Identification in Accordance With A Preferred Embodiment

FIG. 9 is a flow diagram that depicts the hardware and logical flow of control for a device and a software system designed to allow Web-based comparison shopping in conventional, physical, non-Web retail environments. A wireless phone or similar hand-held wireless device 920 with Internet Protocol capability is combined with a miniature barcode reader 910 (installed either inside the phone or on a short cable) and used to scan the Universal Product Code (UPC) bar code on a book or other product 900. The wireless device 920 transmits the bar code via an antennae 930 to the Pocket BargainFinder Service Module (running on a Web server) 940, which converts it to (in the case of books) its International Standard Book Number or (in the case of other products) whatever identifier is appropriate. The Service Module then contacts the appropriate third-party Web site(s) to find price, shipping and availability information on the product from various Web suppliers 950. This information is formatted and displayed on the hand-held device's screen. The IP wireless phone or other hand held device 920 utilizes a wireless modem such as a Ricochet SE Wireless Modem from Metricom. Utilizing this device, a user can hang out in a coffee shop with a portable computer perched on a rickety little table, with a latte sloshing dangerously close to the keyboard, and access the Internet at speeds rivaling direct connect via a telephone line.

The 8-ounce Ricochet SE Wireless Modem is about as large as a pack of cigarettes and setup is extremely simple, simply attach the modem to the back of your portable's screen with the included piece of Velcro, plug the cable into the serial port, flip up the stubby antenna, and transmit. Software setup is equally easy: a straightforward installer adds the Ricochet modem drivers and places the connection icon on your desktop. The functional aspects of the modem are identical to that of a traditional telephone modem.

Of course, wireless performance isn't nearly as reliable as a traditional dial-up phone connection. We were able to get strong connections in several San Francisco locations as long as we stayed near the windows. But inside CNET's all-brick headquarters, the Ricochet couldn't connect at all. When you do get online, performance of up to 28.8 kbps is available with graceful degradation to slower speeds. But even the slower speeds didn't disappoint. Compared to the alternative—connecting via a cellular modem—the Ricochet is much faster, more reliable, and less expensive to use. Naturally, the SE Wireless is battery powered. The modem has continuous battery life of up to 12 hours. And in accordance with a preferred embodiment, we ran down our portable computer's dual cells before the Ricochet started to fade.

Thus, utilizing the wireless modem, a user may utilize the web server software 940 to identify the right product 950 and then use an appropriate device's key(s) to select a supplier and place an order in accordance with a preferred embodiment. The BargainFinder Service Module then consummates the order with the appropriate third-party Web supplier 960.

mySite! Personal Web Site & Intentions Value Network Prototype

mySite! is a high-impact, Internet-based application in accordance with a preferred embodiment that is focused on the theme of delivering services and providing a personalized experience for each customer via a personal web site in a buyer-centric world. The services are intuitively organized around satisfying customer intentions—fundamental life needs or objectives that require extensive planning decisions, and coordination across several dimensions, such as financial planning, healthcare, personal and professional development, family life, and other concerns. Each member owns and maintains his own profile, enabling him to create and browse content in the system targeted specifically at him. From the time a demand for products or services is entered, to the completion of payment, intelligent agents are utilized to conduct research, execute transactions and provide advice. By using advanced profiling and filtering, the intelligent agents learn about the user, improving the services they deliver. Customer intentions include Managing Daily Logistics (e.g., email, calendar, contacts, to-do list, bill payment, shopping, and travel planning); and Moving to a New Community (e.g., finding a place to live, moving household possessions, getting travel and shipping insurance coverage, notifying business and personal contacts, learning about the new community). From a consumer standpoint, mySite! provides a central location where a user can access relevant products and services and accomplish daily tasks with ultimate ease and convenience.

From a business standpoint, mySite! represents a value-added and innovative way to effectively attract, service, and retain customers. Intention value networks allow a user to enter through a personalized site and, and with the assistance of a learning, intelligent agent, seamlessly interact with network participants. An intention value network in accordance with a preferred embodiment provides superior value.

It provides twenty four hour a day, seven days a week access to customized information, advice and products. The information is personalized so that each member views content that is highly customized to assure relevance to the required target user.

Egocentric Interface

An Egocentric Interface is a user interface crafted to satisfy a particular user's needs, preferences and current context. It utilizes the user's personal information that is stored in a central profile database to customize the interface. The user can set security permissions on and preferences for interface elements and content. The content integrated into the Egocentric Interface is customized with related information about the user. When displaying content, the Egocentric Interface will include the relationship between that content and the user in a way that demonstrates how the content relates to the user. For instance, when displaying information about an upcoming ski trip the user has signed up for, the interface will include information about events from the user's personal calendar and contact list, such as other people who will be in the area during the ski trip. This serves to put the new piece of information into a context familiar to the individual user.

FIG. 10A describes the Intention Value Network Architecture implementation for the World Wide Web. For simplification purposes, this diagram ignores the complexity pertaining to security, scalability and privacy. The customer can access the Intention Value Network with any Internet web browser 1010, such as Netscape Navigator or Microsoft Internet Explorer, running on a personal computer connected to the Internet or a Personal Digital Assistant with wireless capability. See FIG. 17 for a more detailed description of the multiple methods for accessing an Intention Value Network. The customer accesses the Intention Value Network through the unique name or IP address associated with the Integrator's Web Server 1020. The Integrator creates the Intention Value Network using a combination of resources, such as the Intention Database 1030, the Content Database 1040, the Supplier Profile Database 1050, and the Customer Profile Database 1060.

The Intention Database 1030 stores all of the information about the structure of the intention and the types of products and services needed to fulfill the intention. Information in this database includes intention steps, areas of interest, layout templates and personalization templates. The Content Database 1040 stores all of the information related to the intention, such as advice, referral information, personalized content, satisfaction ratings, product ratings and progress reports.

The Supplier Profile Database 1050 contains information about the product and service providers integrated into the intention. The information contained in this database provides a link between the intention framework and the suppliers. It includes product lists, features and descriptions, and addresses of the suppliers' product web sites. The Customer Profile Database 1060 contains personal information about the customers, such as name, address, social security number and credit card information, personal preferences, behavioral information, history, and web site layout preferences. The Supplier's Web Server 1070 provides access to all of the supplier's databases necessary to provide information and transactional support to the customer.

The Product Information Database 1080 stores all product-related information, such as features, availability and pricing. The Product Order Database 1090 stores all customer orders. The interface to this database may be

through an Enterprise Resource Planning application offered by SAP, Baan, Oracle or others, or it may be accessible directly through the Supplier's Web Server or application server. The Customer Information Database 1091 stores all of the customer information that the supplier needs to complete a transaction or maintain customer records.

FIG. 10B is a flowchart providing the logic utilized to create a web page within the Egocentric Interface. The environment assumes a web server and a web browser connected through a TCP/IP network, such as over the public Internet or a private Intranet. Possible web servers could include Microsoft Internet Information Server, Netscape Enterprise Server or Apache. Possible web browsers include Microsoft Internet Explorer or Netscape Navigator. The client (i.e. web browser) makes a request 1001 to the server (i.e. web server) for a particular web page. This is usually accomplished by a user clicking on a button or a link within a web page. The web server gets the layout and content preferences 1002 for that particular user, with the request to the database keyed off of a unique user id stored in the client (i.e. web browser) and the User profile database 1003. The web server then retrieves the content 1004 for the page that has been requested from the content database 1005. The relevant user-centric content, such as calendar, email, contact list, and task list items are then retrieved 1006. (See FIG. 11 for a more detailed description of this process.) The query to the database utilizes the user content preferences stored as part of the user profile in the User profile database 1003 to filter the content that is returned. The content that is returned is then formatted into a web page 1007 according to the layout preferences defined in the user profile. The web page is then returned to the client and displayed to the user 1008.

FIG. 11 describes the process of retrieving user-centric content to add to a web page. This process describes 1006 in FIG. 10B in a more detailed fashion. It assumes that the server already has obtained the user profile and the existing content that is going to be integrated into this page. The server parses 1110 the filtered content, looking for instances of events, contact names and email addresses. If any of these are found, they are tagged and stored in a temporary holding space. Then, the server tries to find any user-centric content 1120 stored in various databases. This involves matching the tagged items in the temporary storage space with calendar items 1130 in the Calendar Database 1140; email items 1115 in the Email Database 1114; contact items 1117 in the Contact Database 1168; task list items 1119 in the Task List Database 1118; and news items 1121 in the News Database 1120. After retrieving any relevant user-centric content, it is compiled together and returned 1122.

User Persona

The system allows the user to create a number of different personas that aggregate profile information into sets that are useful in different contexts. A user may create one persona when making purchases for his home. This persona may contain his home address and may indicate that this user is looking to find a good bargain when shopping. The same user may create a second persona that can be used when he is in a work context. This persona may store the user's work address and may indicate that the user prefers certain vendors or works for a certain company that has a discount program in place. When shopping for work-related items, the user may use this persona. A persona may also contain rules and restrictions. For instance, the work persona may restrict the user to making airline reservations with only one travel agent and utilizing booking rules set up by his employer.

FIG. 12 describes the relationship between a user, his multiple personas and his multiple profiles. At the User Level is the User Profile 1200. This profile describes the user and his account information. There is one unique record in the database for each user who has an account. Attached to each user are multiple Personas 1220, 1230 & 1240. These Personas are used to group multiple Profiles into useful contexts. For instance, consider a user who lives in San Francisco and works in Palo Alto, but has a mountain cabin in Lake Tahoe. He has three different contexts in which he might be accessing his site. One context is work-related. The other two are home-life related, but in different locations. The user can create a Persona for Work 1220, a Persona for Home 1230, and a Persona for his cabin home 1240. Each Persona references a different General Profile 1250, 1260 and 1270 which contains the address for that location. Hence, there are three General Profiles. Each Persona also references one of two Travel Profiles. The user maintains a Work Travel Profile 1280 that contains all of the business rules related to booking tickets and making reservations. This Profile may specify, for instance, that this person only travels in Business or First Class and his preferred airline is United Airlines. The Work Persona references this Work Travel Profile. The user may also maintain a Home Travel Profile 1290 that specifies that he prefers to travel in coach and wants to find non-refundable fairs, since they are generally cheaper. Both the Persona for Home and the Persona for the cabin home point to the Home Travel Profile.

FIG. 13 describes the data model that supports the Persona concept. The user table 1310 contains a record for each user who has an account in the system. This table contains a username and a password 1320 as well as a unique identifier. Each user can have multiple Personas 1330, which act as containers for more specialized structures called Profiles 1340. Profiles contain the detailed personal information in Profile Field 1350 records. Attached to each Profile are sets of Profile Restriction 1360 records. These each contain a Name 1370 and a Rule 1380, which define the restriction. The Rule is in the form of a pattern like (if x then y), which allows the Rule to be restricted to certain uses. An example Profile Restriction would be the rule that dictates that the user cannot book a flight on a certain airline contained in the list. This Profile Restriction could be contained in the "Travel" Profile of the "Work" Persona set up by the user's employer, for instance. Each Profile Field also contains a set of Permissions 1390 that are contained in that record. These permissions dictate who has what access rights to that particular Profile Field's information.

Intention-Centric Interface

Satisfying Customer Intentions, such as Planning for Retirement or Relocating requires a specialized interface. Customer Intentions require extensive planning and coordination across many areas, ranging from financial security, housing and transportation to healthcare, personal and professional development, and entertainment, among others. Satisfying Intentions requires a network of complementary businesses, working across industries, to help meet consumers' needs.

An Intention-Centric Interface is a user interface designed to help the user manage personal Intentions. At any given point, the interface content is customized to show only content that relates to that particular Intention. The Intention-Centric Interface allows the user to manage the process of satisfying that particular Intention. This involves a series of discrete steps and a set of content areas the user can access. At any point, the user can also switch the interface to manage a different Intention, and this act will

change the content of the interface to include only that content which is relevant to the satisfaction of the newly selected Intention.

FIG. 14 provides a detailed description of the data model needed to support an Intention-Centric Interface. Each User Persona 1410 (see FIG. 13 for a more detailed description of the Persona data model.) has any number of active User Intentions 1420. Each active User Intention is given a Nickname 1430, which is the display name the user sees on the screen. Each active User Intention also contains a number of Data Fields 1440, which contain any user data collected throughout the interaction with the user. For instance, if the user had filled out a form on the screen and one of the fields was Social Security Number, the corresponding Data Field would contain Name="SSN" 1450, Value="999-99-9999" 1460. Each User Intention also keeps track of Intention Step 1470 completion status. The Completion 1480 field indicates whether the user has completed the step. Every User Intention is a user-specific version of a Generic Intention 1490, which is the default model for that Intention for all users. The Generic Intention is customized through Custom Rules 1411 and 1412 that are attached to the sub-steps in the Intention. These Custom Rules are patterns describing how the system will customize the Intention for each individual user using the individual user's profile information.

Statistical Agent

An agent keeps track of key statistics for each user. These statistics are used in a manner similar to the Tamagochi virtual reality pet toy to encourage certain behaviors from the user. The statistics that are recorded are frequency of login, frequency of rating of content such as news articles, and activity of agents, measured by the number of tasks which it performs in a certain period. This information is used by the system to emotionally appeal to the user to encourage certain behaviors.

FIG. 15 describes the process for generating the page that displays the agent's current statistics. When the user requests the agent statistics page 1510 with the client browser, the server retrieves the users' statistics 1520 from the users' profile database 1530. The server then performs the mathematical calculations necessary to create a normalized set of statistics 1540. The server then retrieves the formulas 1550 from the content database 1560 that will be used to calculate the user-centric statistics. Graphs are then generated 1570 using the generic formulas and that user's statistics. These graphs are inserted into a template to create the statistics page 1580. This page is then returned to the user 1590.

Personalized Product Report Service

The system provide Consumer Report-like service that is customized for each user based on a user profile. The system records and provides ratings from users about product quality and desirability on a number of dimensions. The difference between this system and traditional product quality measurement services is that the ratings that come back to the users are personalized. This service works by finding the people who have the closest match to the user's profile and have previously rated the product being asked for. Using this algorithm will help to ensure that the product reports sent back to the user only contain statistics from people who are similar to that user.

FIG. 16 describes the algorithm for determining the personalized product ratings for a user. When the user requests a product report 1610 for product X, the algorithm retrieves the profiles 1620 from the profile database 1630 (which includes product ratings) of those users who have

previously rated that product. Then the system retrieves the default thresholds 1640 for the profile matching algorithm from the content database 1650. It then maps all of the short list of users along several dimensions specified in the profile matching algorithm 1660. The top *n* (specified previously as a threshold variable) nearest neighbors are then determined and a test is performed to decide if they are within distance *y* (also specified previously as a threshold variable) of the user's profile in the set 1670 using the results from the profile matching algorithm. If they are not within the threshold, then the threshold variables are relaxed 1680, and the test is run again. This processing is repeated until the test returns true. The product ratings from the smaller set of *n* nearest neighbors are then used to determine a number of product statistics 1690 along several dimensions. Those statistics are inserted into a product report template 1695 and returned to the user 1697 as a product report.

Personal Profile and Services Ubiquity

This system provides one central storage place for a person's profile. This storage place is a server available through the public Internet, accessible by any device that is connected to the Internet and has appropriate access. Because of the ubiquitous accessibility of the profile, numerous access devices can be used to customize services for the user based on his profile. For example, a merchant's web site can use this profile to provide personalized content to the user. A Personal Digital Assistant (PDA) with Internet access can synchronize the person's calendar, email, contact list, task list and notes on the PDA with the version stored in the Internet site. This enables the person to only have to maintain one version of this data in order to have it available whenever it is needed and in whatever formats it is needed.

FIG. 17 presents the detailed logic associated with the many different methods for accessing this centrally stored profile. The profile database 1710 is the central storage place for the users' profile information. The profile gateway server 1720 receives all requests for profile information, whether from the user himself or merchants trying to provide a service to the user. The profile gateway server is responsible for ensuring that information is only given out when the profile owner specifically grants permission. Any device that can access the public Internet 1730 over TCP/IP (a standard network communications protocol) is able to request information from the profile database via intelligent HTTP requests. Consumers will be able to gain access to services from devices such as their televisions 1740, mobile phones, Smart Cards, gas meters, water meters, kitchen appliances, security systems, desktop computers, laptops, pocket organizers, PDAs, and their vehicles, among others. Likewise, merchants 1750 will be able to access those profiles (given permission from the consumer who owns each profile), and will be able to offer customized, personalized services to consumers because of this.

One possible use of the ubiquitous profile is for a hotel chain. A consumer can carry a Smart Card that holds a digital certificate uniquely identifying him. This Smart Card's digital certificate has been issued by the system and it recorded his profile information into the profile database. The consumer brings this card into a hotel chain and checks in. The hotel employee swipes the Smart Card and the consumer enters his Pin number, unlocking the digital certificate. The certificate is sent to the profile gateway server (using a secure transmission protocol) and is authenticated. The hotel is then given access to a certain part of the consumer's profile that he has previously specified. The hotel can then retrieve all of the consumer's billing information as well as preferences for hotel room, etc. The hotel

can also access the consumer's movie and dining preferences and offer customized menus for both of them. The hotel can offer to send an email to the consumer's spouse letting him/her know the person checked into the hotel and is safe. All transaction information can be uploaded to the consumer's profile after the hotel checks him in. This will allow partners of the hotel to utilize the information about the consumer that the hotel has gathered (again, given the consumer's permission).

Intention Value Network

In an Intention Value Network, the overall integrator system coordinates the delivery of products and services for a user. The integrator manages a network of approved suppliers providing products and services, both physical and virtual, to a user based on the user's preferences as reflected in the user's profile. The integrator manages the relationship between suppliers and consumers and coordinates the suppliers' fulfillment of consumers' intentions. It does this by providing the consumer with information about products and suppliers and offering objective advice, among other things.

FIG. 18 discloses the detailed interaction between a consumer and the integrator involving one supplier. The user accesses a Web Browser 1810 and requests product and pricing information from the integrator. The request is sent from the user's browser to the integrator's Web/Application Server 1820. The user's preferences and personal information is obtained from an integrator's customer profile database 1830 and returned to the Web/Application server. The requested product information is extracted from the supplier's product database 1840 and customized for the particular customer. The Web/Application server updates the supplier's customer information database 1850 with the inquiry information about the customer. The product and pricing information is then formatted into a Web Page 1860 and returned to the customer's Web Browser.

Summary Agent

A suite of software agents running on the application and web servers are programmed to take care of repetitive or mundane tasks for the user. The agents work according to rules set up by the user and are only allowed to perform tasks explicitly defined by the user. The agents can take care of paying bills for the user, filtering content and emails, and providing a summary view of tasks and agent activity. The user interface for the agent can be modified to suit the particular user.

FIG. 19 discloses the logic in accordance with a preferred embodiment processing by an agent to generate a verbal summary for the user. When the user requests the summary page 1900, the server gets the user's agent preferences 1920, such as agent type, rules and summary level from the user profile database 1930. The server gets the content 1940, such as emails, to do list items, news, and bills, from the content database 1950. The agent parses all of this content, using the rules stored in the profile database, and summarizes the content 1960. The content is formatted into a web page 1970 according to a template. The text for the agent's speech is generated 1980, using the content from the content database 1990 and speech templates stored in the database. This speech text is inserted into the web page 1995 and the page is returned to the user 1997.

Trusted Third Party

The above scenario requires the web site to maintain a guarantee of privacy of information according to a published policy. This system is the consumer's Trusted Third Party, acting on his behalf in every case, erring on the side of privacy of information, rather than on the side of stimulation of commerce opportunities. The Trusted Third Party has a

set of processes in place that guarantee certain complicity with the stated policy.
"meCommerce"

This word extends the word "eCommerce" to mean "personalized electronic commerce." FIG. 20 illustrates a display login in accordance with a preferred embodiment. The display is implemented as a Microsoft Internet Explorer application with an agent 2000 that guides a user through the process of interacting with the system to customize and personalize various system components to gather information and interact with the user's personal requirements. A user enters a username at 2010 and a password at 2020 and selects a button 2040 to initiate the login procedure. As the logo 2030 suggests, the system transforms electronic commerce into a personalized, so called "me" commerce.

FIG. 21 illustrates a managing daily logistics display in accordance with a preferred embodiment. A user is greeted by an animated agent 2100 with a personalized message 2190. The user can select from various activities based on requirements, including travel 2110, household chores 2120, finances 2130 and marketplace activities 2140. Icons 2142 for routine tasks such as e-mail, calendaring and document preparation are also provided to facilitate rapid navigation from one activity to another. Direct links 2146 are also provided to allow transfer of news and other items of interest. Various profiles can be selected based on where the user is located. For example, work, home or vacation. The profiles can be added 2170 as a user requires a new profile for another location. Various items 2180 of personal information are collected from the user to support various endeavors. Moreover, permissions 2150 are set for items 2180 to assure information is timely and current.

FIG. 22 illustrates a user main display in accordance with a preferred embodiment. World 2200 and local news 2210 is provided based on a user's preference. The user has also selected real estate 2230 as an item to provide direct information on the main display. Also, a different agent 2220 is provided based on the user's preference.

FIG. 23 illustrates an agent interaction in accordance with a preferred embodiment. The agent 2310 is communicating information 2300 to a user indicating that the user's life insurance needs have changed and pointing the user to the chart that best summarizes the information for the user. Particular tips 2395 are provided to facilitate more detailed information based on current user statistics. A chart 2370 of the user's life insurance needs is also highlighted at the center of the display to assist the user in determining appropriate action. A button 2380 is provided to facilitate changing the policy and a set of buttons 2390 are provided to assist a user in selecting various views of the user's insurance requirements.

Event Backgrounder

An Event Backgrounder is a short description of an upcoming event that is sent to the user just before an event. The Event Backgrounder is constantly updated with the latest information related to this event. Pertinent information such as itinerary and logistics are included, and other useful information, such as people the user knows who might be in the same location, are also included. The purpose of the Event Backgrounder is to provide the most up-to-date information about an event, drawing from a number of resources, such as public web sites and the user's calendar and contact lists, to allow the user to react optimally in a given situation. Vicinity Friend Finder

This software looks for opportunities to tell the user when a friend, family member or acquaintance is or is going to be in the same vicinity as the user. This software scans the

user's calendar for upcoming events. It then uses a geographic map to compare those calendar events with the calendar events of people who are listed in his contact list. It then informs the user of any matches, thus telling the user that someone is scheduled to be near him at a particular time.

Information Overload

The term information overload is now relatively understood in both its definition as well as its implications and consequences. People have a finite amount of attention that is available at any one time, but there is more and more vying for that attention every day. In short, too much information and too little time are the primary factors complicating the lives of most knowledge workers today.

The first attempts to dynamically deal with information overload were primarily focused on the intelligent filtering of information such that the quantity of information would be lessened. Rather than simply removing random bits of information, however, most of these approaches tried to be intelligent about what information was ultimately presented to the user. This was accomplished by evaluating each document based on the user's interests and discarding the less relevant ones. It follows, therefore, that the quality was also increased.

Filtering the information is only a first step in dealing with information in this new age. Arguably, just as important as the quality of the document is having ready access to it. Once you have entered a meeting, a document containing critical information about the meeting subject delivered to your office is of little value. As the speed of business continues to increase fueled by the technologies of interconnectedness, the ability to receive quality information wherever and whenever you are becomes critical. This new approach is called intelligent information delivery and is heralding in a new information age.

A preferred embodiment demonstrates the intelligent information delivery theory described above in an attempt to not only reduce information overload, but to deliver high quality information where and when users' require it. In other words, the system delivers right information to the right person at the right time and the right place.

Active Knowledge Management System Description

FIG. 24 is a block diagram of an active knowledge management system in accordance with a preferred embodiment. The system consists of the following parts: back-end 2400 connection to one or more servers, personal mobile wireless clients (Awareness Machine) 2430, 2436, public clients (Magic Wall) 2410, 2420, web clients 2446, 2448, e-mail clients 2450, 2460.

Back-end Server (2400) Processes

FIG. 25 is a block diagram of a back end server in accordance with a preferred embodiment. The back-end (2400 of FIG. 24) is a computer system that has the following software active: Intelligent Agents Coordinator (Munin) 2580, Information Prioritization Subsystem 2530, a set of continuously and periodically running information gathering and processing Intelligent Agents 2500, 2502 and 2504, User Profiles Database 2542 and supporting software, Information Channels Database 2542 and supporting software, communications software 2550, information transformation software 2560, and auxiliary software.

The Awareness Machine (2446 & 2448 of FIG. 24)

The Awareness Machine is a combination of hardware device and software application. The hardware consists of handheld personal computer and wireless communications device. The Awareness Machine reflects a constantly updated state-of-the-owner's-world by continually receiving a wireless trickle of information. This information, mined

and processed by a suite of intelligent agents, consists of mail messages, news that meets each user's preferences, schedule updates, background information on upcoming meetings and events, as well as weather and traffic.

FIG. 26 is a block diagram of a magic wall in accordance with a preferred embodiment.

The Magic Wall

The Magic Wall hardware includes:

Computer system 2640 connected to the back-end server

Sensor array 2634, 2630 and 2632 detects presence, position, and identity of a person

Large touch-sensitive display 2620

Sound input 2610/output 2614 hardware

The Magic Wall software supports:

Multimedia output compatible with current Web standards

Speech recognition

Tactile input

Intelligent agents representations in the form of speech-enabled animated characters

The Magic Wall operates as follows:

1. If a user appears in the vicinity of Magic Wall, the sensor array triggers "user here" event that sends an environmental cue containing the person's id and the location to the Intelligent Agent Coordinator.

2. User is identified based on the information returned by the sensor array.

3. The Magic Wall switches to "locked on the user" mode. If another user approaches, the system will notify him or her that it cannot serve another user while the current user is being served.

4. Intelligent Agent Coordinator is notified about the user presence.

5. The Intelligent Agent Coordinator decides if there is pertinent information that the user would be interested in and Magic Wall location time-sensitive information to show (e.g. traffic report, meeting reminder). If such information exists, it is prepared for delivery. If not, control is transferred to the Information Prioritization Subsystem.

6. Information Prioritization Subsystem decides what information is most relevant to the user based on their personal profile, freshness of the information, and the Intelligent Agent Coordinator's prior suggestions.

7. The page of information identified as the most relevant to the user at this time and place is shown. The act of the information delivery can also include animation and speech output of the intelligent agent representation.

8. If a user desires, he or she can ask Magic Wall to show a particular page. The Magic Wall recognizes the speech fragment and then identifies and shows the requested page.

9. As the user departs from the Magic Wall area, the sensor array triggers "user left" event.

10. The Magic Wall switches back to the waiting state.

Other Clients

The Web client is a standard browser navigating to a set of Web pages which allow user to see the same information that is available via the Magic Wall.

The e-mail client is any standard e-mail program.

Intelligent Agent Coordinator Description

This piece of code is the coordinating agent (or meta-agent) for the Active Knowledge Management system. This means that all communications between the system and each user, as well as communication between the different minion agents are handled (coordinated) by the Intelligent Agent Coordinator. Examples of these minion agents are:

BackgroundFinder—an agent that parses meeting text determining important keywords and phrases and finds background information on the meeting for each user

TrafficFinder—an agent that finds traffic information for each user based on where they live

Several other agents that are responsible for doing statistical analysis of the data in each user's profile and updating fields pertinent to that data

The Intelligent Agent Coordinator 2580 of FIG. 25 is also the user's "interface" to the system, in that whenever the user interacts with the system, regardless of the GUI or other end-user interface, they are ultimately dealing with (asking questions of or sending commands to) the Intelligent Agent Coordinator. The Intelligent Agent Coordinator has four primary responsibilities: 1) monitoring user activities, 2) handling information requests, 3) maintaining each user's profile, and 4) routing information to and from users and to and from the other respective agents.

Monitoring User Activities

Anytime a user triggers a sensor the Intelligent Agent Coordinator receives an "environmental cue." These cues not only enable the Intelligent Agent Coordinator to gain an understanding where users' are for information delivery purposes, but also to learn the standard patterns (arrival time, departure time, etc.) of each persons' life. These patterns are constantly being updated and refined in an attempt to increase the system's intelligence when delivering information. For instance, today it is not uncommon for a person to have several email accounts (work-based, home-based, mobile-based, etc.) as well as several different computers involved in the retrieval process for all of these accounts. Thus, for the Intelligent Agent Coordinator to be successful in delivering information to the correct location it must take into account all of these accounts and the times that the user is likely to be accessing them in order to maximize the probability that the user will see the information. This will be discussed further in another section.

Handling Information Requests

The Intelligent Agent Coordinator handles information requests from other agents in order to personalize information intended for each user and to more accurately reflect each user's interests in the information they are given. These requests will commonly be related to the user's profile. For instance, if an agent was preparing a traffic report for a user it may request the traffic region (search string) of that user from the Intelligent Agent Coordinator. All access to the user's profile data is accessed in this method.

Maintaining User Profiles

User profiles contain extensive information about the users. This information is a blend of user-specified data and information that the Intelligent Agent Coordinator has learned and extrapolated from each user's information and activities. In order to protect the data contained in the profiles, the Intelligent Agent Coordinator must handle all user information requests. The Intelligent Agent Coordinator is constantly modifying and updating these profiles by watching the user's activities and attempting to learn the patterns of their lives in order to assist in the more routine, mundane tasks. The Intelligent Agent Coordinator also employs other agents to glean meaning from each user's daily activities. These agents mine this data trying to discover indications of current interests, long-term interests, as well as time delivery preferences for each type of information. Another important aspect of the Intelligent Agent Coordinator's observations is that it also tries to determine where each user is physically located throughout the day for routing purposes.

Information Routing

Most people are mobile throughout their day. The Intelligent Agent Coordinator tries to be sensitive to this fact by

attempting to determine, both by observation (unsupervised learning) and from cues from the environment, where users are or are likely to be located. This is certainly important for determining where to send the user's information, but also for determining in which format to send the information. For instance, if a user were at her desk and using the web client, the Intelligent Agent Coordinator would be receiving indications of activity from her PC and would know to send any necessary information there. In addition, because desktop PCs are generally quite powerful, a full-featured, graphically intense version could be sent. However, consider an alternative situation: the Intelligent Agent Coordinator has received an indication (via the keycard reader next to the exit) that you have just left the building. Minutes later the Intelligent Agent Coordinator also receives notification that you have received an urgent message. The Intelligent Agent Coordinator, knowing that you have left the building and having not received any other indications, assumes that you are reachable via your handheld device (for which it also knows the capabilities) and sends the text of the urgent message there, rather than a more graphically-oriented version.

Inherent Innovations

The Active Knowledge Management system represents some of the most advanced thinking in the world of knowledge management and human computer interaction. Some of the primary innovations include the following:

The Intelligent Agent Coordinator as illustrated above.

The development, demonstration, and realization of the theory of Intelligent Information Delivery

Support for several channels of information delivery, all of which utilize a common back-end. For instance, if a user is in front of a Magic Wall the information will be presented in a multimedia-rich form. If the system determines that the user is mobile, the information will be sent by to their Awareness Machine in standard text. It facilitates delivery of information whenever and wherever a user requires the information.

Personalization of information based not only on a static user profile, but also by taking into account history of the user interactions and current real-time situation including "who, where, and when" awareness.

Utilization of fast and scalable Information Prioritization Subsystem that takes into account Intelligent Agents Coordinator opinion, user preferences, and history of user interactions. It takes the load of mundane decisions off the Intelligent Agents part therefore allowing the agents to be much more sophisticated and precise without compromising the system scalability.

Speech recognition and speech synthesis in combination with intelligent agent animated representation and tactile input provides for efficient, intuitive, and emotionally rewarding interaction with the system.

Supporting Code in Accordance With A Preferred Embodiment

The following code is written and executed in the Microsoft Active Server Pages environment in accordance with a preferred embodiment. It consists primarily of Microsoft Jscript with some database calls embedded in the code to query and store information in the database.

Intention-Centric Interface

Create an Intention ASP Page ("intention_create.asp")

```
<%@ LANGUAGE = "JScript" %>
<%
Response.Buffer = true;
Response.Expires = 0;
%>
<html>
<head>
<title>Create An Intention</title>
</head>
<body bgcolor="#FFE9D5" style="font-family: Arial" text="#000000">
<%
//Define some variables
upl = Server.CreateObject("SoftArtisans.FileUp")
intention_name = upl.Form("intention_name")
intention_desc = upl.Form("intention_desc")
//intention_name = Request.Form("intention_name")
//intention_desc = Request.Form("intention_desc")
//intention_icon = Request.Form("intention_icon")
submitted = upl.Form("submitted")
items = new Enumerator(upl.Form)
%>
<%
//Establish connection to the database
objConnection = Server.CreateObject ("ADODB.Connection")
ObjConnection.Open ("Maelstrom")
%>
<%
//Check to see if the person hit the button and do the appropriate thing
if (submitted == "Add/Delete")
{
    flag = "false"
    //loop through all the inputs
    while(!items.atEnd())
    {
        i = items.item()
        //if items are checked then delete them
    }
}
```

-continued

Intention-Centric Interface

```

        if(upl.Form(i) == "on")
        {
            objConnection.Execute("delete from user_intention where
intention_id = " + i);
            objConnection.Execute("delete from intentions where intention_id = "
+ i);
            objConnection.Execute("delete from tools_to_intention where
intention_id = " + i)
            flag = "true"
        }
        items.moveToNext( )
    }
    // if items were not deleted then insert whatever is in the text field in the
database
    if(flag == "false")
    {
        intention_name_short = intention_name.replace(/ /g, " ")
        objConnection.Execute("INSERT INTO intentions
(intention_name,intention_desc,intention_icon) values(' " + intention_name + " ',' ' +
intention_desc + " ',' ' " + intention_name_short + ".gif" + " ')"
        Response.write("the intention short name is " + intention_name_short);
        upl.SaveAs("E:/development/asp_examples/" + intention_name_short + ".gif")
    }
}

// Query the database to show the most recent items.
rsCustomersList = objConnection.Execute("SELECT * FROM intentions")

%>
<input type="Submit" name="return_to_mcp" value="Go to Main Control Panel"
onclick="location.href='default.asp'" >
<form method="post" action="intention_create.asp" enctype="multipart/form-data" >
<TABLE border=0>
<tr><td colspan="2"><font face="Arial" size="+1"><b>Enter in a new
intention</b></font></td></tr>
<tr><td><font face="Arial">Name:</font></td><td><INPUT TYPE="text"
name="intention_name"></td></tr>
<tr><td><font face="Arial">Description:</font></td><td><TEXTAREA
name="intention_desc"></TEXTAREA></td></tr>
<tr><td><font face="Arial">Icon Image:</font></td><td><INPUT TYPE="file"
NAME="intention_icon" size=40></td></tr>
<tr><td colspan="2"><INPUT type="submit" name="submitted" value="Add/Delete"></td></tr>
</TABLE>
<HR>
<font face="Arial" size="+1"><b>Current Intentions</b></font>
<TABLE>
<tr bgcolor=#E69780 align="center">
<td>
<font color="white">Delete</font>
</td>
<td>
<font color="white">Intention</font>
</td>
<td>
<font color="white">Description</font>
</td>
<td>
<font color="white">Image</font>
</td>
</tr>
</table>
<%
// Loop over the intentions in the list
counter = 0;
while (!rsCustomersList.EOF)
{
%>
<tr bgcolor="white" style="font-size: smaller">
<td align="center">
<INPUT type="checkbox" name=" "><%=rsCustomersList ("intention_id")%>
</td>
<td>
<%= rsCustomersList ("intention_name")%>
</td>
<td>
<%= rsCustomersList ("intention_desc")%>
</td>
<td>
<%= rsCustomersList("intention_icon")%>

```

-continued

Intention-Centric Interface

```

        </td>
    </tr>
    <%
    counter++
    rsCustomersList.MoveNext ()
    %>
    </TABLE>
    <br>
    Available Tools
    </form>
    </BODY>
    </HTML>

    Retrieve Intentions List ASP Page ("intentions_list.asp")

    <!-- #include file="include/check_authentication.inc" -->
    <HTML>
    <HEAD>
        <TITLE>mySite! Intentions List</TITLE>
    <SCRIPT LANGUAGE="JavaScript">
        function intentionsList () {
            this.internalArray = new Array( );
            <%
            // establish connection to the database
            objConnection = Server.CreateObject ("ADODB.Connection");
            objConnection.Open("Maelstrom");
            // create query
            intentionsQuery = objConnection.Execute("SELECT * FROM intentions ORDER BY
            intention_name asc");
            %>
            // write out the options
            <%
            numOptions = 0
            while (!intentionsQuery.EOF) {
                intentionName = intentionsQuery("intention_name");
                intentionIcon = intentionsQuery("intention_icon");
            %>

                this.internalArray[<%= numOptions%>] = new Array(2);
                this.internalArray[<%= numOptions%>][0] = "<%= intentionName %>";
                this.internalArray[<%= numOptions%>][1] = "images/<%= intentionIcon
            %>";
            <%
                numOptions++; intentionsQuery.moveNext( );
            <%
            }
            }
            numIntentions = <%= numOptions%>;
            intentionArray = new intentionsList( ).internalArray;
            function selectIntention () {
                for (i=0;i<numIntentions;i++) {
                    if (IntentionsListSelect.options[i].selected) {
                        intentionNameTextField.value = intentionArray[i][0];
                        //intentionPicture.src = intentionArray[i][1];
                        break;
                    }
                }
            }
        }
    </SCRIPT>
    </HEAD>
    <BODY BGCOLOR="<%=Session("main_background")%>" style="font-family: Arial">
    <CENTER>
        <!-- <FORM NAME="intention_list"> -->
        <TABLE FRAME="BOX" border=0 CELLSPACING="2" CELLSPACING="2">
        <TR><TD COLSPAN="3" STYLE="font: 20pt arial" ALIGN="CENTER"><B>Add a mySite!
        Intention</B></TD></TR>
        <TR><TD COLSPAN="3">&nbsp;</TD></TR>
        <TR>
            <TD width="100"><font size="-1">Please Select An Intention You Would Like to Add
            to Your List</font></TD>
            <TD colspan=2>
                <SELECT ID="IntentionsListSelect" NAME="IntentionsListSelect" SIZE="10"
                style="font: 9pt Arial;" onClick="selectIntention()">
                <%
                intentionsQuery.moveFirst ( );
                for(j=0;j<numOptions;j++) {
                    <OPTION VALUE="<%= intentionsQuery("intention_id") %>" <% if (j ==

```

Intention-Centric Interface

Display User Intention List ASP Page (excerpted from "navigation.asp")

```
<DIV ID="intentionlist" style="position: absolute; width:210; height:95; left: 365pt; top:
-5; visibility: hidden; font-family: Arial; font-color: #000000; font: 8pt Arial; " >
<DIV style="position: absolute; top:7; left:7; height:78; width:210; z-index:2;
background: <%=Session("main_background")%>; border: solid 1pt #000000; padding: 3pt;
overflow: auto; alink: black; link: black;">
<boby LINK="#000000" ALINK="#000000" vlink="black">

// create query
intentionsQuery = objConnection.Execute("SELECT user_intention."
FROM user_intention, user_intention_to_persona WHERE
user_intention_to_persona.user_persona_id = " & Session("currentUserPersona") + " AND
user_intention_to_persona.user_intention_id = user_intention.user_intention_id");
numintentions = 0;
Response.Write ("<SCRIPT>numintentions=" +
intentionsQuery.RecordCount + "</SCRIPT><TABLE cellpadding='0' width='100%'
cellspacing='0'>");

while (!intentionsQuery.EOF)
{
%>
<TR><TD><a href="javascript:changeIntention('<%=
intentionsQuery("user_intention_id") %>', '<%=numintentions%>')"
onmouseover="mouseOverTab ()" onmouseout="mouseOutOfTab()" "><font color="Black" face="arial"
size="-2"><%= intentionsQuery("intention_custom_name") %></font></a></TD><TD><IMG
align="right" SRC="images/delete.gif" alt="Delete this intention"
onClick="confirmDelete (<%=intentionsQuery("user_intention_id") %>) "></TD></TR>
<%=numintentions++; intentionsQuery.moveNext(); %>
%>
{
Response.Write("<SCRIPT>numintentions=" + numintentions
+ "</SCRIPT>");
%>
<tr><td colspan="2"><hr></td></tr>
<TR><td colspan="2"><a href="javascript:changeIntention('add
...', '<%=numintentions%>');" onmouseover="mouseOverTab()"
onmouseout="mouseOutOfTab()" "><font color="Black" face="arial" size="-2">add
... </font></a></td></TR>
</table>

</body>
</DIV>
<DIV style="position: absolute; top:0; left:-5; width: 230; height:105; z-index:1; "
onmouseover="intentionlist.style.visibility='hidden' "
```

-continued

Intention-Centric Interface

```
onmouseout="intentionlist.style.visibility='hidden' "
onmouseover="intentionlist.style.visibility='hidden' "></DIV>
</DIV>
</DIV>
```

While various embodiments have been described above, it should be understood that they have been presented by way of example only, and not limitation. Thus, the breadth and scope of a preferred embodiment should not be limited by any of the above described exemplary embodiments, but should be defined only in accordance with the following claims and their equivalents.

What is claimed is:

1. A method for creating a user network interface which is accessible from a plurality of locations, comprising the steps of:

identifying a user;

identifying in a database a plurality of stored profiles corresponding to the user;

determining which of the identified plurality of stored profiles corresponds to a current location of the user;

identifying information of interest to the user based on the determined user profile;

prioritizing the information of interest to the user based on the determined user profile;

displaying the information of interest formatted on a web page in accordance with the determined user profile; and

updating the determined user profile in the database based on interaction with the displayed information by the user in accordance with the current location of the user.

2. A method for creating a user network interface as recited in claim 1, wherein each profile has a unique intention associated therewith.

3. A method for creating a user network interface as recited in claim 1, wherein each profile has a unique address associated therewith.

4. A method for creating a user network interface as recited in claim 1, wherein each profile has unique restrictions associated therewith.

5. A method for creating a user network interface as recited in claim 1, wherein each profile has unique business rules associated therewith.

6. A method for creating a user network interface as recited in claim 1, wherein the information conveyed to the user is identified in one or more databases selected from the group consisting of a content database, a calendar database, an electronic email database, a contact database, a task database, and a news database.

7. A method for creating a user network interface as recited in claim 1, including the step of monitoring a user's activities to develop and store in the user's profile a time-based probability pattern of the user's location, and when the location of the user is not identified, delivering the information of interest to at least the location having the highest probability the user will view the information.

8. A method for creating a user network interface as recited in claim 7, including the step of formatting information for delivery to the user based on characteristics of the user's current environment.

9. A method for creating a user network interface as recited in claim 1, including the step of automatically

sensing a user proximate to a publicly accessible multimedia display device, identifying the user and displaying on the multimedia display device the information of interest to the identified user.

10. A method for creating a user network interface as recited in claim 1, wherein the information of interest has a plurality of categories, and the method includes the step of updating the determined user profile with a time delivery preference for each category of information of interest based on said interactions.

11. A method for creating a user network interface as recited in claim 10, wherein the information of interest is displayed in accordance with the time delivery preference in the determined user profile.

12. A method for creating a user network interface as recited in claim 1, wherein the information of interest is further based on a response to a query developed from a description of an upcoming event retrieved from a calendar application for the user.

13. An apparatus for creating a user network interface which is accessible from a plurality of locations, comprising:

a processor;

a memory that stores information under the control of the processor;

logic that identifies a user;

logic that identifies in a database a plurality of stored profiles corresponding to the user

logic that determines which of the identified plurality of stored profiles corresponds to a current location of the user;

logic that identifies information of interest to the user based on the determined user profile;

logic that prioritizes the information of interest to the user based on the determined user profile;

logic that displays the information of interest on a web page formatted in accordance with the determined user profile; and

logic that updates the determined user profile in the database based on interaction with the displayed information by the user in accordance with the current location of the user.

14. A computer-readable medium for storing program code that, when executed, creates a user network interface which is accessible from a plurality of locations, the stored program code providing execution of the method steps of:

identifying a user;

identifying in a database a plurality of stored profiles corresponding to the user;

determining which of the identified plurality of stored profiles corresponds to a current location of the user;

identifying information of interest to the user based on the determined user profile;

prioritizing the information of interest to the user based on the determined user profile;

49

displaying the information of interest formatted on a web page in accordance with the determined user profile; and

updating the determined user profile in the database based on interaction with the displayed information by the user in accordance with the current location of the user.

15 15. A computer-readable medium as recited in claim 14, wherein each profile has a unique intention associated therewith.

10 16. A computer-readable medium as recited in claim 14, wherein each profile has a unique address associated therewith.

17. A computer-readable medium as recited in claim 14, wherein each profile has unique restrictions associated therewith.

18. A computer-readable medium as recited in claim 14, wherein each profile has unique business rules associated herewith.

19. A computer-readable medium as recited in claim 14, wherein the information of interest to the user is identified in one or more databases selected from the group consisting of a content database, a calendar database, an electronic email database, a contact database, a task database, and a news database.

50

20. A computer-readable medium as recited in claim 14, including a code segment embodied on a computer-readable medium that monitors a user's activities, a code segment that stores in the user's profile a time-based probability pattern of the user's location based in the user's monitored activities, and when the location of the user is not identified, a code segment that delivers the information of interest to at least the location having the highest probability the user will view the information.

21. A computer-readable medium as recited in claim 14, including a code segment embodied on a computer-readable medium that formats information for delivery to the user based on characteristics of the user's current environment.

15 22. A computer-readable medium as recited in claim 14 including a code segment embodied on a computer-readable medium that automatically senses a user proximate to a publicly accessible multimedia display device, a code segment that identifies the user and a code segment that displays on the multimedia display device the information of interest to the identified user.

* * * * *